TENSIONAL ANALYSIS OF FUNCTIONALLY GRADED MATERIALS USING BOUNDARY INTEGRAL EQUATIONS

Grupo de Investigación
MECÁNICA DE SÓLIDOS Y ESTRUCTURAS

Tesis Doctoral

TENSIONAL ANALYSIS OF FUNCTIONALLY GRADED MATERIALS USING BOUNDARY INTEGRAL EQUATIONS

Autor:
MIGUEL ANGEL RIVEIRO TABOADA

Director:
RAFAEL GALLEGO SEVILLA

Marzo de 2014

Departamento de Mecánica de Estructuras e Ingeniería Hidráulica
Universidad de Granada

ugr | Universidad
de **Granada**

Universidad de Granada
E.T.S. Ingenieros de Caminos, Canales y Puertos
Departamento de Mecánica de Estructuras e Ingeniería Hidráulica
Campus de Fuentenueva, s/n

El doctorando MIGUEL ANGEL RIVEIRO TABOADA y el director de la tesis RAFAEL GALLEGO SEVILLA. Garantizamos, al firmar esta tesis doctoral, que el trabajo ha sido realizado por el doctorando bajo la dirección del director de la tesis y hasta donde nuestro conocimiento alcanza, en la realización del trabajo, se han respetado los derechos de otros autores a ser citados, cuando se han utilizado sus resultados o publicaciones.
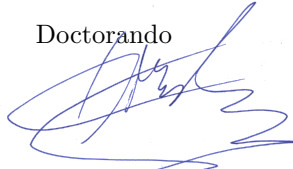
Granada, Marzo de 2014.

Director de la Tesis   Doctorando

Fdo.: RAFAEL GALLEGO SEVILLA   MIGUEL ANGEL RIVEIRO TABOADA

To my father
who no longer can read it

Prologue

This thesis, titled " Tensional analysis of Functionally Graded Materials using Boundary Integral Equations" is part of the work related to the Boundary Element Method, carried out by the research group led by Professor Rafael Gallego Sevilla, the advisor of this thesis, whom I thank for his guidance and help throughout the years necessary for its development, because this work would not have not possible without him.

Also thank all the department of Structural Mechanics and Hydraulic Engineering of the University of Granada, by so openly accepted me and provide me a work environment with peers, who have long ceased to be regarded as colleagues, and now I consider great friends. I especially want to thank my fellow office for so many years, Inas Faris, for having supported and stoically endured my unconventional way of programming.

Thank my family, who have suffered the requiremets in time and effort of this kind of work. My parents, who supported me from the distance and were always an example and a mirror to see myself. My sister, my brother-in-law and my nieces who are always waiting back to see his uncle. My stepfamily, who has treated me like one of the family from the start. My wife and my child, that make it all worthwhile and I love more than I can express.

<div align="right">Miguel Angel Riveiro Taboada</div>

Granada

March of 2014

Resumen

En esta tesis, se presenta la aplicación de un algoritmo basado en el Método de Elementos de Contorno (BEM), para la resolución de problemas elastostáticos en dominios tridimensionales que incluyan materiales inhomogéneos, como los Functionally Graded Materials (FGM).

La utilización del Método de la ecuación Análoga nos permite transformar el operador diferencial del problema original en otro, con un término independiente desconocido, pero cuya solución fundamental sea conocida. Mediante esta transformación y el uso combinado del Método de Elementos de Contorno, la aproximación del término desconocido mediante funciones de base radial y el Método de Doble Reciprocidad, se obtiene un sistema de Ecuaciones Integrales de Contorno en función de los desplazamientos y sus flujos. La aplicación del operador diferencial del problema original y, las condiciones de contorno que incluyan derivadas de los desplazamientos, proporciona las ecuaciones adicionales que permiten calcular los coeficientes que definen el término independiente desconocido y los flujos de los desplazamientos. La construcción de estos nuevos grupos de ecuaciones es expuesto junto al análisis de las nuevas singularidades que surgen en este contexto. El carácter de contorno del método se mantiene en el sentido de que el dominio de integración de las ecuaciones resultantes se limita al contorno, ya que las funciones de base radial se escogen de tal forma que la ecuación análoga puede ser resuelta de forma analítica. La extensión a multidominios incluyendo el acoplamiento de esta metodología con el Método de Elementos de Contorno estándar también es analizado.

La implementación de este algoritmo se realiza mediante programación orientada objetos, con el objetivo de generar un código de elementos de contorno altamente escalable, reutilizable y mantenible. La liberación del nuevo estándar de FORTRAN 2003 y la disponibilidad de compiladores que soporten las nuevas características de este lenguaje, permite el desarrollo de un nueva generación de códigos BEM en FORTRAN. En esta tesis se estudia la aplicabilidad de estas nuevas características, de cara al diseño de un programa BEM global que pueda integrar de forma segura y eficiente las

diferentes metodologías BEM.

Varios ejemplos numéricos de problemas lineales elastostáticos en tres dimensiones, que involucren materiales inhomogéneos tipo FGM se adjuntan para la validación del algoritmo presentado, incluyendo problemas multidominio en combinación con el Método de Elementos de Contorno estándar. Se incluyen estudios de convergencia y análisis comparativos de comportamiento de diferentes familias de funciones de aproximación.

Abstract

In this dissertation the application of a Boundary Element Method (BEM) based algorithm to elastostatic problems, involving 3D non-homogeneous materials like Functionally Graded Materials (FGMs) is presented.

The Analog Equation Method (AEM) is used to transform the original problem into a new problem with unknown fictitious source but known Fundamental Solution. By means of this transformation a system of uncoupled Boundary Integral Equations (BIEs) depending on displacements and fluxes is first obtained, combining standard Boundary Element discretization, Radial Basis Functions (RBFs) approximation for the fictitious source and the Dual Reciprocity Method. The application of the original differential operator and the boundary conditions involving derivatives of the displacement, provides additional equations to compute the unknown fictitious source and the flux of the displacements. The construction of these new groups of equations is exposed besides the analysis of the new singularities that arise in this context. The boundary character of the method is maintained since the integrals involved in the equations are limited only to the boundary due to the RBFs are selected in such a way that the corresponding analog equation could be solved analytically. The extension of this AEM-BEM methodology to multidomains including the coupling with standard Boundary Element Method schemes is also analyzed.

In order to implement these algorithm an object oriented programming style have been chosen to implement a highly scalable, reusable and maintainable Boundary Element Method code. The release of the new 2003 FORTRAN standard and the availability of compilers capable to support the new features like encapsulation, inheritance, and polymorphism has opened the way to a new generation of FORTRAN BEM codes. A study of the new supported features and its use to design a global BEM program in an object oriented style, able to easily scale and integrate different techniques, is presented.

Several numerical examples for three-dimensional problems in continuously

non-homogeneous, isotropic and linear elastic FGMs are presented to validate the algorithm, including multidomain problems coupled with standard Boundary Element Method. Convergence studies and comparative analysis of the behavior of different families of approximating functions are also included.

Contents

CONTENTS

List of Figures

List of Tables

Part I

Preliminaries

All men by nature desire to know.

Aristotle

1

Introduction

1.1 Motivation

Competitiveness, productivity and cost reduction have become commonly used terminology, even for the general public. The discard of still valuable resources and the increase of the performances that we can obtain, have always been a key point in the strategy of industries and governments, but this fact it is more true nowadays.

From the structural point of view we can translate these ideas to the need of better materials, better designs and better maintenance. In the field of materials, development of new solutions has brought us the evolution of the materials starting from pure monolithic materials like aluminum to alloys such as steel, composite materials such as carbon fiber and more recently the emergence of Functionally Graded Materials (FGMs) (in Figure 1.1 it can be observed the evolution in terms of composition in different families of Boeing aircrafts). What underlies this evolution is that the right combination of different materials with different properties produces materials with higher performances that the original ones.

In recent years there has been an increased interest in the study of Functionally Graded Materials. In this new class of advanced materials, there is a continuous variation of the internal microstructure along the geometry
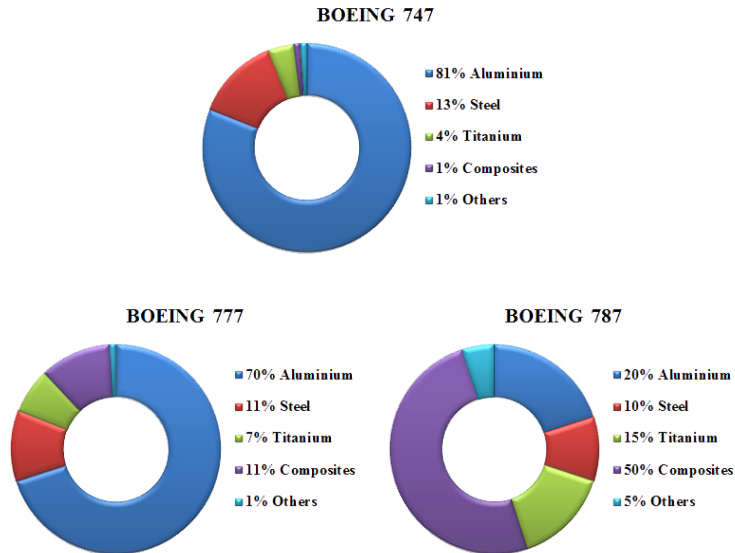
Figure 1.1: Evolution of materials in Boeing aircraft models

of the material, resulting in a continuous variation of the properties at macroscopic level. This allows to design materials whose properties are adapted to the requirements and avoid interface problems displayed on the multiphase materials due to discrete jumps of properties.

The Boundary Element Method (BEM) is a numerical methodology capable of solving problems defined by systems of partial differential equations, which has been widely used in recent decades to solve many problems of scientific and industrial interest. As will be explained later, the standard BEM formulation is restricted, in practice, to problems involving domains with constant properties.

In recent years, the extension of this family of numerical techniques to produce algorithms able to solve problems involving these new materials, has been explored by the scientific community. In the literature, it can be found several methodologies that, maintaining the pure boundary character, in the sense that it is not necessary to mesh the domain and that the domain

of integration is limited to the boundary, can be applied to these problems.

The study of these techniques has led us to consider that the combination of the Analog Equation Method (AEM), introduced by Katsikadelis [60], with the BEM methodology was the most advantageous for their potential generality, but it has, in its current state of development, several limitations such as it has only been implemented in 2D and it does not allow generic boundary conditions without the use of finite differences.

On the other hand, the multiplicity of BEM codes which are individually developed to implement each new algorithm, even within our own research group, suggested that it was more than convenient to streamline efforts to create a single scalable software base, which avoids the dispersion codes and works.

The emergence of new compilers capable of handling the latest iteration of FORTRAN and its new object-oriented capabilities, allows to establish a framework to develop this unique platform integrating the previously developed codes.

1.2   Objectives

Two sets of objectives have been raised in this dissertation:

1) Generalize the AEM-BEM formulation including

- General analog operators.

- Build an integral formulation that allows to input directly boundary conditions derived from the main variables.

- Characterize this formulation to elastic problems

- Implement this methodology to 3D problems, performing numerical validation tests and convergence analysis of the resulting scheme.

- Perform a comparative study of the different families of approximation functions under the AEM-BEM framework.

- Develop the coupling formulation with the Standard Boundary Element Method and validate it by numerical tests.

2) Building the core of an object-oriented program based on the latest iteration of FORTRAN (2003/2010) for, besides the implementation of this algorithm, easily scale and integrate, in a natural way, different BEM techniques. The initial defined requirements of our code are

- Multi-Domain.

- Multi-Space (2D/3D).

- Multi-Physics: several kinds of problems supported including scalar or vector variables.

- Able to use different integration schemes at element level.

- Able to use different BEM algorithms.

- Extensive use of dynamic memory (writes data once / reads many).

- Multiple discretization schemes including simultaneous use of different interpolations, elements.

- Support different kinds of boundary conditions including inter-phases.

- Reduction of the computational cost.

1.3 Thesis Organization

This work is divided into three main parts.

Part I covers the first three chapters. In the first introductory chapter the motivations and objectives of this work have been exposed.

Chapter 2 reviews previous works and the state of the art. It contains a literature review of the algorithms based on the Boundary Element Method capable of solve problems involving materials with varying properties such as FGM (from a purely descriptive point of view). Four groups of methodologies have been identified including the selected one that forms the basis for this work.

In Chapter 3 some previous preliminary concepts that will be used throughout this work are described. It includes a summary of the Boundary Element Method formulation, both the development of the Boundary Integral Equation (BIE) as the discretization process using a generalized scheme. Subsequently, a similar treatment is made of Dual Reicprocity Method. Then there is a brief analysis of functional approximation schemes in the context of Dual reciprocity Methods. Finally, the Analog Equation Method, which is the basis for this work, is introduced, both in its original formulation, and in its later development combined with dual reciprocity methods.

Parts II and III include the original contributions of this work. Part II comprised chapters 4 through 7 and includes the theoretical developments of the methodology used in this dissertation.

In Chapter 4 the limitations of the original AEM-BEM methodology are outlined and a set of integral equations is formulated using a generic analog operator. After the discretization of this system of equations, any boundary value problems for linear partial differential equations with generic boundary conditions can be solved. Also, a generic discretization scheme in order to obtain the discretized linear algebraic system of equations is exposed.

In Chapter 5 two simplifications are introduced. The first one is due to the choice of Laplace operator as the analog operator. The second one is due to the choice of elastic problem to be solved. Anisotropic and isotropic elastic formulation are covered.

Chapter 6 will focus on the discretization process and the assembly of the system of algebraic equations that can solve the inhomogeneous isotropic

elastic problems studied in this work. Next post-processing procedures are discussed, other numerical aspects are reviewed and AEM-BEM coupling with the standard BEM methodology is analyzed.

The design of object-oriented program based on the latest iteration of FORTRAN (2003/2010) is presented in Chapter 7 for, besides the implementation of the algorithm presented in this paper, build a scalable software to integrate, in a natural way, different BEM techniques.

Part III comprised chapters 8 and 9, including the numerical results and conclusions.

Examples of validation AEM-BEM algorithm are included in Chapter 8. Convergence analysis of problems with Dirichlet and mixed boundary conditions are attached, including comparatives of several types of approximation functions. Additionally two examples of the AEM-BEM coupling with the standard BEM methodology are enclosed.

The conclusions and future works are presented in Chapter 9.

Finally an appendix chapter including several mathematical developments that, by its length, have been separated from the main text to improve its readability.

Reason itself does not work instinctively, but requires trial, practice, and instruction in order gradually to progress from one level of insight to another.

Immanuel Kant

2

State of the Art

The previous chapter introduced the motivation and objectives of this thesis, in relation to the elastic study of Functionally Graded Materials using algorithms based on the Boundary Element Method.

The study of these kind of problems have been addressed using different techniques. As a general reference, Birman and Byrd published, in 2007, a literature review of the state of the art [17], relative to the FGM modeling and analysis and more recently (2013) Jha, Kant and Singh [52] published another review focused on thermoelastic analysis and vibration of FGM plates.

This chapter includes a brief literature review of precedents, focusing exclusively on Boundary Element techniques, being outside the scope of this chapter, works based on finite elements (see e.g. [64] or [1]), behavior models ([102], [122]) or other approaches. Is also not delve into a bibliographic study of the development of the BEM basic methodology [1]. Several reviews of the mathematical foundations and historical background [26], textbooks that systematize basic principles and applications (e.g. [18], [19], [118] or [5]) and analysis of the development of some BEM variants [120] are available in the literature.

---

[1]In the following chapters it will presented in more detail the basics of the Boundary Element Method and the other algorithms used in this work.

Although the aim of this thesis focuses on the elastic problem, its analysis and implementation using Boundary Element Methods, is integrated into the general extension of these methods to inhomogeneous materials with varying properties. The available solutions, including the one used in this thesis, are generally applicable, at least conceptually, to other type of problems.

The key feature of the Boundary Element Method, in its conventional formulation, is that only boundary discretization is required in order to solve the problem. This is attained by the use of a fundamental solution that satisfies the problem system of equations when the loading is a concentrated source. Thus, even though research conducted in recent years have dramatically expand the scope of application of BEM, including, inter alia, acoustics [4], contact mechanics [3] or soil-structure interaction [50], a Fundamental Solution is still needed in analytical form or with low computational cost.

In practice, this condition often limits the application of the Boundary Element Method to linear systems with constant coefficients. The emergence of new materials including Functionally Graded Materials (FGMs) has increased the interest in boundary element techniques capable of dealing with materials showing such non-homogeneous properties. Several approaches have been reported in the literature to overcome these difficulties. Focusing on this aspect, a non comprehensive list of BEM variants comprising four main groups can be constructed.

2.1 Pure Multidomain Methods

The first approach to solve problems involving inhomogeneous materials is through the use of standard formulations and multidomain schemes. This type of approach is present from the beginning of BEM development although it is limited to piece-wise homogeneous problems.

The multidomain techniques divided the domain into multiple zones, each characterized by a homogeneous material with constant properties. Standard

Boundary Element method is applied in every zone and coupling equations are added in the interphase to obtain the complete system of equations. The resulting matrix is a band matrix at the scale of the blocks derived from the domains. These techniques were early introduced (see [11] or [19]). Several studies have been divulged, focus on the study of assembly techniques and resolution of these algebraic system of equations defined by the matrix by means of factorization (as in [31]) ,or static condensation of degrees of freedom as in [53]) among others[2]. It can be noted, finally, that this approach is not only applied to collocation schemes but also to Galerkin schemes [67] or [24].

2.2 Fundamental Solution Calculation and Change of Variables

In this section, we include studies of obtaining fundamental solutions and techniques that transform the original problem to one whose fundamental solution is known.

The first type of problems to be studied in this context was potential problems in inhomogeneous solids. Chen [27] studied one and two dimensional problems governed by the equation

$$\nabla \cdot (k(\mathbf{x}) \nabla \phi) = 0 \qquad (2.1)$$

By changes of variable, it can be shown that it is possible to obtain the fundamental solution of these problems providing that

$$\nabla^2 k^{\frac{1}{2}} = 0 \qquad (2.2)$$

---

[2]Several works like [97], have shown that, for problems with a large amount of degrees of freedom, the multidomain approaches are superior in terms of computational efficiency and numerical conditioning comparing to a single-zone scheme.

Later, Shaw and Makris [104] collect these ideas, and apply these schemes based on changes of variables to Laplace and Helmholtz problems. Shaw [105] performed a systematic study also using transformations of the independent variable. Ang, Clements and Kusuma [7] generalized the applicability of these schemes to two-dimensional cases where $K(\mathbf{x}) = X(x)Y(y)$ if some additional conditions are fulfilled.

Clements obtained solutions for some specific cases of second-order elliptic equations of the type [29]

$$\frac{\partial}{\partial x_j}\left[a_{ijkl}(x_1,x_2)\frac{\partial \phi_k}{\partial x_l}\right] + b_{ik}(x_1,x_2)\phi_k = 0 \qquad (2.3)$$

and [28]

$$\frac{\partial}{\partial x_j}\left[a_{ijkl}(\mathbf{x}))\frac{\partial \phi_k}{\partial x_l}\right] = 0 \qquad (2.4)$$

Manolis and Shaw [73] obtained particular solutions for the elastodynamic equation for some specific properties variation. These constraints result in a Poisson ratio of 0.25, a quadratic variation of the Young modulus in a single coordinate and a density variation proportional to the Young modulus.

Sutradhar, Paulino and Gray [108] used similar techniques to obtain the Green function for the diffusion problem

$$\nabla \cdot (k\nabla\phi) = c\frac{\partial \phi}{\partial t} \qquad (2.5)$$

for cases in which the thermal conductivity k and the specific heat c have an exponential variation in a single coordinate of the type $k = k_0 e^{2\beta z}$

Gray, Kaplan, Richardson and Paulino [48] obtained the fundamental solution of the potential problem, in two and three dimensions, in the case of an exponential variation in a single coordinate of the k coefficient $k = k_0 e^{-2i\alpha z}$, where the coefficient α may be imaginary. Berger, Martin, Mantic and

Gray [15] extended this approach to cover anisotropic solids with the same type of property variation where, in this case, the k coefficient becomes a symmetric matrix. Kuo and Chen [65] obtained, using a different approach, the fundamental solution for both the potential and the diffusion problem in anisotropic solids with an exponential variation of properties.

Through Fourier transforms, Martin, Richardson, Gray and Berger obtained the fundamental solution of the three-dimensional elastic problem [77], for the case where the material has an exponential variation in a single coordinate of Lame coefficients

$$\lambda = \lambda_0 e^{2\boldsymbol{\beta}\mathbf{x}} \qquad \mu = \mu_0 e^{2\boldsymbol{\beta}\mathbf{x}} \qquad (2.6)$$

where $\boldsymbol{\beta}$ is a constant vector that indicate the direction of the properties variation. This fundamental solution is not given in explicit form, but has terms as integrals (this solution was corrected and evaluated against problems with analytic solution [30]). Chan, Gray, Kaplan and Paulino [22] obtained the fundamental solution for the two-dimensional version of the previous problem, for the same type of materials. This solution is also non-explicit and has a term in the form of one-dimensional Fourier integrals.

Sutradhar and Paulino changed the approach of the methods that use changes of variable introducing the so called "Simple BEM". Instead of obtaining a fundamental solution of the problem with non-homogeneous materials, the problem itself is transformed into a homogeneous one, so standard available algorithms and its implementations could be used with minor variations because this method simply introduces changes in the boundary conditions of the resultant problem. Using these techniques, it is possible to solve, in case of materials whose properties vary along quadratic, exponential or trigonometric functions in one coordinate, potential problems [107] and diffusion problems [110] where the thermal conductivity k and the specific heat c variation are proportional. This method has been used by the same authors to study potential problems with multiple cracks [94] involving solids with the same properties variation as the previous case.

In figure (2.1) an example (taken from [110]) of the application of this technique is presented, which is compared with the results obtained using finite elements, showing the high accuracy of the BEM algorithms.
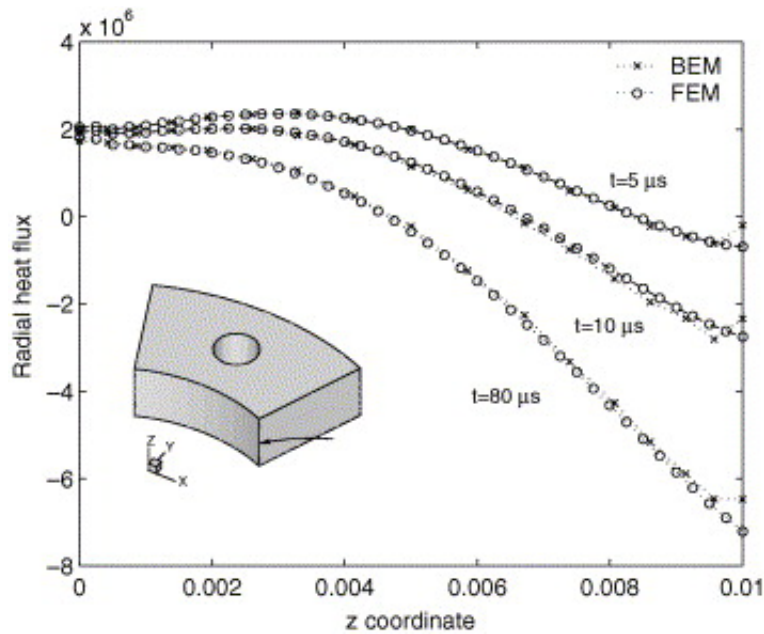
Figure 2.1: Radial heat flux along the interior edge [110]

All the works named in this section have the advantage of allowing the reuse of existing codes, needing only routines associated with the new fundamental solutions [3], or in the case of "Simple BEM" only modifying the boundary conditions of the problem. In addition they can be used in collocation or Galerkin schemes.

---

[3]In practice, it is clear that it is necessary to make further modifications, for example, such as those associated with boundary conditions.

2.3 Methodologies that use domain integrals

Generally, the application fields of the methodologies outlined in the previous section are strongly limited to a small range of functional variation of properties, despite the advantage that the fundamental solution is obtained in analytical form. Although they have practical applications they lack of generality. Since the late 80s, it was proposed as a general solution a set of methods, which can be grouped conceptually because they introduce domain integrals in the formulation. The best known subfamilies are called dual reciprocity methods (see as reference [93]), multiple reciprocity methods [90] or radial integration methods [41] among others.

The general idea behind these methods is to split the differential operator which governs the problem in two parts. The first part can be treated using the standard BEM methodology and the remaining terms are grouped into volume integrals in the second part[4].

The key feature of these methods is the transformation of the domain integrals into boundary integrals or sums of approximation functions. Originally, this type of algorithms were designed to solve problems where the independent term was a general function, but they have also been applied successfully to problems involving inhomogeneous materials. Ang, Clements and Vahdati [6] studied differential problems of elliptic type involving inhomogeneous anisotropic solids with general properties variation. In this case the differential operator is

$$\frac{\partial}{\partial x_i}\left[\lambda_{ij}\frac{\partial u}{\partial x_j}\right] = 0 \quad \text{in} \quad \mathbb{R}^2 \qquad (2.7)$$

where $\lambda_{ij}(x,y) = \lambda_{ij}^0 g(x,y)$

Marin, Elliott, Heggs, Ingham and Lesnic studied Helmholtz type problems [75], with general variations of the k coefficient in two dimensions domains

---

[4]In the following chapters the dual reciprocity method is explained in detail. At this point the descriptions are limited to conceptual ideas.

and Fahmy focused on magneto-thermo-viscoelastic problems involving anisotropic inhomogeneous solids [36].

Subsequently Gao, Guo and Zhang combined radial integration techniques with multidomain methods to solve elastic problems in two and three dimensional domains [42], where the Poisson coefficient ν is constant and the shear modulus G is defined by a general function. In the Figure 2.2, an example of a three-dimensional problem for an exponential variation of G, where eight subdomains have been used, is provided. A comparison of the BEM technique with a the Finite Element Method is shown.

(a) BEM model of a cuboid consisting in eight subdomains

(b) Numerical results for transversal inhomogeneous material
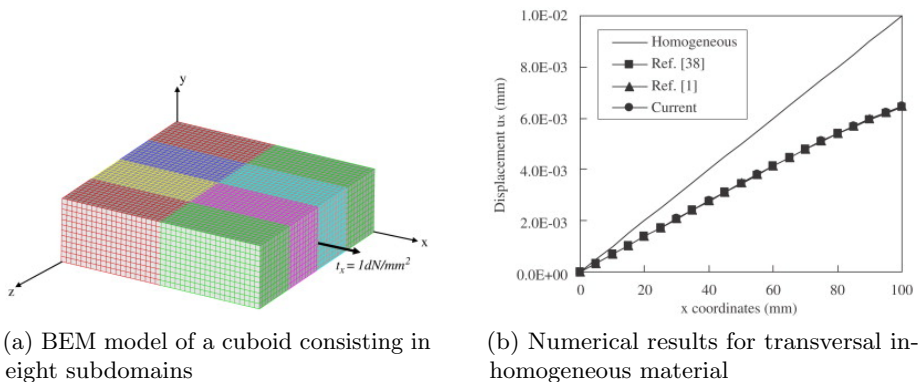
Figure 2.2: Example taken from [42]

This same technique was subsequently used in the study of thermoelasticity [40], fracture [123] and diffusion [95].

The last technique presented in this section is the one that has been selected as a starting point in this thesis. This methodology was introduced by Katsikadelis [57] for solving boundary value problems, involving linear or non linear second order differential operators. The Analog Equation Method and the Boundary Element Method are combined to produce an algorithm whose fundamental solution is independent of the problem. The key idea is to replace the original differential operator by the analog operator (which

has known fundamental solution) and treating the unknown term produced by this replacement using techniques that avoid domain discretization.

In the original algorithm, the unknown term is estimated directly by summation of approximation functions (mainly radial basis functions). It can be found in the literature several works studying buckling of plates [87], dynamic analysis of nonlinear membranes [59] or dynamic analysis of composite steel-concrete structures [103] among others. Subsequently Katsikadelis [60] replace the original evaluation of the unknown term with a volume integral, combining the previous formulation with the Dual Reciprocity Method.

Although the next chapter will detail the basis of this methodology, notice that the original form of this formulation implied that the boundary conditions of the problems must be functions of the problem variables and its fluxes. Nerantzaki and Kandilas applied this methodology to anisotropic elastic problems [86], where the boundary conditions are given in tensions, evaluating the tangential derivatives needed to close the problem by means of a finite differences approach.

In this work this idea will be extended and the analytical derivative of the integral equations is employed in order to deal with any type of boundary conditions, without using additional numerical approximations.

2.4 Other Methodologies

There are other approaches in the literature that can extend the range of application of the Boundary Element Method for treating non-homogeneous materials (and nonlinear problems). Liao introduced in the late 90's the so called General Boundary Element Method General (see eg [70], [68] or [69]). In this approach, a perturbation is introduced into the equation and an expansion series is used, resulting in an iterative scheme where the residue decreases gradually. This approach allows to cover linear and non-linear problems, but the computational cost can be high due to the need of the iterative calculation. Figure 2.3 shows an example of application

of this method to a fluid mechanics problem. In this case, a two-dimensional problem involving an incompressible at moderately high Reynolds is studied.
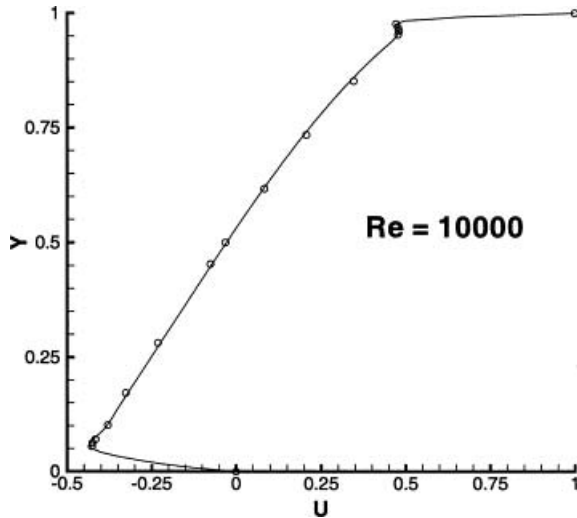
Figure 2.3: Results taken from [119]

Finally two *meshless* methodologies will be mentioned. Formally they are not Boundary Element type techniques, but they have emerged partly in response to the limitations of its applicability and are close related to it. The first method is called the Local Boundary Integral Equation Method (LBIE) introduced by Zhu, Zhang and Atluri [124] and the second one is the Boundary Knot Method (BKM) introduced by Chen and Tanaka [25].

3

Preliminary Concepts

3.1 Boundary Element Method

The Boundary Element Method (BEM) includes a family of numerical techniques able to solve problems defined by systems of linear partial differential equations formulated as sets of integral equations.

Although this general definition of the Boundary Element Method can be also used to refer to different families of numerical techniques, it allows us to establish a first division with respect the techniques which methodology is based on the direct use of the differential formulation (strong formulation) of the problem. Historically, the Finite Difference Method (FDM) [81], which is based on the direct discretization of the differential equations, can be pointed out as the main precursor of this family and, even now, it is still been used successfully to deal with the solution of a large number of physical or engineering problems. Recently the methods based on strong formulations have been received an important boost with the development of meshfree (or meshless) methodologies which use the strong or weak formulation of the problem. In the first group we can mention, by way of example, the Method of Fundamental solutions (MFS) [37] or the Radial basis functions collocation method (RBFCM) [54], [38] among others.

In contrast to the methods based on the differential formulation, methodolo-

gies based on integral formulations (the so-called "weak formulation") can be found in the literature as it has been mentioned. Without being exhaustive, we can mention, for example, the Finite Volume Method (FVM) [115], the Finite Element Method (FEM) [125], several meshfree type methodologies like the meshless local Petrov-Galekin (MLPG) method [10] and the Boundary Element Method (BEM) [19]. The last method gets its name because under certain conditions, the mesh procedure and the corresponding integrals involved in this numerical technique are limited to the boundary of the domain. This key feature is due to the use of the so-called fundamental solution used as a function of weight for the construction of the integral formulation. In contrast, FEM type numerical methods use weight functions that, broadly speaking, are independent of the problem and focus on accurate approximation of the problem variables. In BEM, the weight function is related to the problem to solve and, consequently, added "a priori" information that allows several simplifications in the integral formulation.

3.1.1 Potential Integral Equation

Although, in next sections (see for example (3.4)) variations of the standard BEM formulation capable of solving problems involving inhomogeneous materials as Functionally Graded Materials (FGM)[1] will be analyzed in detail, this section will focus on a brief review of the Boundary Element Method in its traditional direct formulation.

Selecting a problem governed by the Laplace operator, as the starting point of the analysis, a short summary of the procedure is presented. This type of problems have been extensively studied by the potential theory and they have a wide range of applications including steady heat transfer, electrostatics, ideal fluid flow and more (several examples can be found in [118]). The direct formulation, based on Green's identities, is employed to obtain the integral formulation in preference to indirect formulations

---

[1]In the sense arbitrarily inhomogeneous, as in the literature are available fundamental solutions applicable to problems involving materials with several variations of properties.

based on potentials[2].

Although this formulation is focused on the Laplace problem, and, in general, the resulting integral equations varies depending on the differential operator governing the problem, the procedure is completely analogous to other type of problems.
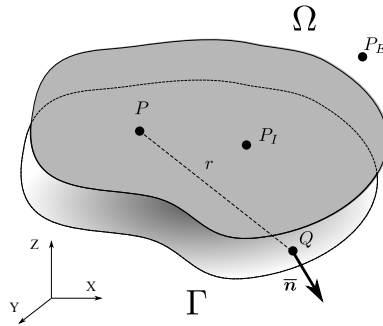
Figure 3.1: A volume Ω bounded by a close surface Γ

Let be a general potential problem formulation defined by:

$$\nabla \cdot (\mathbf{K}(\mathbf{x}) \nabla \phi) + b(\mathbf{x}) = 0 \qquad \boxed{3.1}$$

It is assumed that an isotropic and homogeneous material is studied, so $\mathbf{K}(\mathbf{x}) = \mathbf{I} \, k_0$. Without loss of generality, the problem can be simplified assuming $k_0 = 1$. Introducing these simplifications, equation (3.1) can be reduced to the so-called Poisson equation. In this case, $\phi(x, y, z)$ is a scalar function defined in a three-dimensional space so

$$\nabla^2 \phi = \nabla \bullet \nabla \phi = \sum_{i=1}^{3} \left(\frac{\partial^2 \phi}{\partial x_i^2} \right) = -b(\mathbf{x}) \qquad \boxed{3.2}$$

---

[2]The so-called single layer potential and double layer potential.

where function $\phi(\mathbf{x})$ has a domain of definition defined by the region of space Ω bounded by a close surface named Γ (see figure 3.1 for details). The function $\phi(\mathbf{x})$ is called potential function and its corresponding surface flux is

$$\frac{\partial \phi}{\partial \mathbf{n}} = \nabla \phi \bullet \mathbf{n} \tag{3.3}$$

The formulation of the integral equation of the problem denoted by (3.2), is based on the use of the Method of Weighted Residuals. With this in mind, we can write that

$$\int_{\Omega} w(\mathbf{x}) \left[\nabla^2 \phi(\mathbf{x}) + b(\mathbf{x}) \right] d\Omega = 0 \tag{3.4}$$

for any sufficiently well behaved function $w(\mathbf{x})$ defined in the domain Ω.

By means of the divergence theorem is easy to obtain the so-called Green's identities of Green[3].

Green's first identity

$$\int_{\Omega} \nabla w(\mathbf{x}) \nabla \phi(\mathbf{x}) \, d\Omega = \int_{\Gamma} w(\mathbf{x}) \frac{\partial \phi}{\partial \mathbf{n}}(\mathbf{x}) \, d\Gamma + \int_{\Omega} w(\mathbf{x}) b(\mathbf{x}) \, d\Omega \tag{3.5}$$

Green's second identity[4]

$$\int_{\Gamma} \left(\phi(\mathbf{x}) \frac{\partial w}{\partial \mathbf{n}}(\mathbf{x}) - w(\mathbf{x}) \frac{\partial \phi}{\partial \mathbf{n}}(\mathbf{x}) \right) d\Gamma = \int_{\Omega} \left(w(\mathbf{x}) b(\mathbf{x}) + \nabla^2 w(\mathbf{x}) \phi(\mathbf{x}) \right) d\Omega \tag{3.6}$$

To obtain the third Green identity is necessary to particularize $w = w^*$ in expression (3.6) by using a special type of function called Fundamental

---

[3]The details of these transformations are omitted in this work for the sake of brevity. They can be found in many textbooks, such as [118].

[4]In Solid Mechanics the equivalent expression is the theorem of reciprocity.

Solution. In the case of a problem governed by Laplace operator 3.1 the fundamental solution satisfies

$$\nabla^2 w^* + \delta\left(\mathbf{x} - \mathbf{z}\right) = 0 \qquad (3.7)$$

Combining the previous expression with equation (3.6) we obtain, for $\mathbf{z} \in \Omega$

Green's third identity

$$\phi\left(\mathbf{z}\right) + \int_\Gamma \left(\phi\left(\mathbf{x}\right) \frac{\partial w^*}{\partial \mathbf{n}}\left(\mathbf{x}; \mathbf{z}\right) - w^*\left(\mathbf{x}; \mathbf{z}\right) \frac{\partial \phi}{\partial \mathbf{n}}\left(\mathbf{x}\right) \right) \mathrm{d}\Gamma = \int_\Omega w^*\left(\mathbf{x}; \mathbf{z}\right) b\left(\mathbf{x}\right) \mathrm{d}\Omega \qquad (3.8)$$

It's clear that the fundamental solution depends on the differential operator that governs the problem to be solved. In the case of the Laplace operator in a three dimensional space the solution of (3.7) is available in many textbooks as [19]

$$w^* = \frac{1}{4\pi r} \qquad \frac{\partial w^*}{\partial \mathbf{n}} = -\frac{r_{,i} n_i}{4\pi r^2} \qquad (3.9a)$$

$$\text{where} \quad r = \|\mathbf{x} - \mathbf{z}\| \quad r_{,i} = \frac{r_i}{r} \quad r_i = x_i - z_i \qquad (3.9b)$$

It has previously been pointed out that, in the Boundary Element Method, under certain conditions, the integrals that appear in the formulation, and, consequently, the associated process of discretization and meshing are restricted to the boundaries of the domain. If equation (3.8) is analyzed it is clear that if the independent term of equation $b\left(\mathbf{x}\right) = 0$, the domain integral vanishes and a pure boundary integral equation is obtained. In this case, equation (3.1) is known as Laplace equation, and (3.8) is reduced to

$$\phi\left(\mathbf{z}\right) = \int_\Gamma \left(w^*\left(\mathbf{x}; \mathbf{z}\right) \frac{\partial \phi}{\partial \mathbf{n}}\left(\mathbf{x}\right) - \phi\left(\mathbf{x}\right) \frac{\partial w^*}{\partial \mathbf{n}}\left(\mathbf{x}; \mathbf{z}\right) \right) \mathrm{d}\Gamma \qquad (3.10)$$

When $\mathbf{z} \notin \Omega$ it is clear that $\delta\left(\mathbf{x} - \mathbf{z}\right) = 0$ for every point in the domain, and it is easy to show that equation (3.10) is reduced to
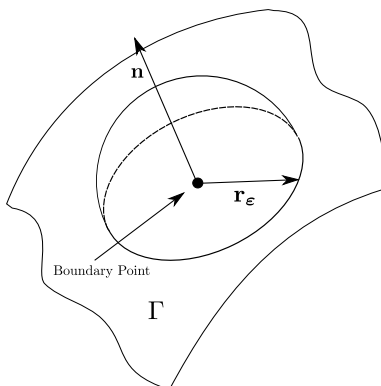
Figure 3.2: Hemisphere around a boundary point at $\mathbf{z}$

$$\int_\Gamma \left(\phi\left(\mathbf{x}\right) \frac{\partial w^*}{\partial \mathbf{n}}\left(\mathbf{x};\mathbf{z}\right) - w^*\left(\mathbf{x};\mathbf{z}\right) \frac{\partial \phi}{\partial \mathbf{n}}\left(\mathbf{x}\right) \right) \mathrm{d}\Gamma = 0 \qquad \boxed{3.11}$$

This equation is known as the *exterior* integral equation.

3.1.2 Integral Equation on a Boundary point

As it has been pointed out previously, equation (3.8) is valid for $\mathbf{z} \in \Omega$. If the pole $\mathbf{z}$ of equation is set in the boundary the fundamental solution of the problem, defined in (3.9a), have a singularity on $\mathbf{z} = \mathbf{x}$. In this case, a detailed analysis of the convergence of the integral equation must be performed. The whole process is reviewed in several textbooks available in the literature (see for example [19]).

In this thesis a brief review of this procedure is included. The key idea is to introduce a deformation in the domain, so that the collocation on $\mathbf{z}$, which originally was placed on the boundary, becomes an interior point. The way to do this is augmenting the domain by a hemisphere of radius ϵ centered on $\mathbf{z}$, as shown in Figure (3.2).

The next step is the decomposition of the integral into two parts. The first includes the entire domain except the hemisphere of radius ε and its value coincides with the Cauchy Principal value when $\varepsilon \to 0$. The second part, corresponding to the hemisphere, is often called free term and its value is calculated analytically. Thus, in general

$$\int_{\Gamma} f \mathrm{d}\Gamma = \lim_{\varepsilon \to 0} \left(\int_{\Gamma - \Gamma_{\varepsilon}} f \mathrm{d}\Gamma \right) + \lim_{\varepsilon \to 0} \left(\int_{\Gamma_{\varepsilon}} f \mathrm{d}\Gamma \right) = \fint_{\Gamma} f \mathrm{d}\Gamma + \text{free term} \qquad \boxed{3.12}$$

In the case of the three-dimensional Laplace operator, the details of the limiting process are included in the Appendix of this thesis. For a point $\mathbf{z} \in \Gamma$, it can be proved

$$\left(1 - \frac{\Delta\Omega(\mathbf{z})}{4\pi} \right) \phi(\mathbf{z}) = \fint_{\Gamma} \left(w^*(\mathbf{x}; \mathbf{z}) \frac{\partial \phi}{\partial \mathbf{n}}(\mathbf{x}) - \phi(\mathbf{x}) \frac{\partial w^*}{\partial \mathbf{n}}(\mathbf{x}; \mathbf{z}) - \right) \mathrm{d}\Gamma \qquad \boxed{3.13}$$

where the variable $\Delta\Omega(\mathbf{z})$ have been introduced. This variable represents the solid angle of the domain at the point $(\mathbf{z})$. In the case of a smooth surface, this value, as it can be easily checked, is 2π and, consequently

$$1 - \frac{\Delta\Omega(\mathbf{z})}{4\pi} = \frac{1}{2} \qquad \boxed{3.14}$$

3.1.3 General Integral Potential Equation

Taking into account the above results, we can formulate a general equation for any position of $\mathbf{z}$

$$c(\mathbf{z}) \phi(\mathbf{z}) + \int_{\Gamma} \left(\phi(\mathbf{x}) \frac{\partial w^*}{\partial \mathbf{n}}(\mathbf{x}; \mathbf{z}) - w^*(\mathbf{x}; \mathbf{z}) \frac{\partial \phi}{\partial \mathbf{n}}(\mathbf{x}) \right) \mathrm{d}\Gamma \qquad \boxed{3.15}$$

where

$$c\left(\mathbf{z}\right) = \begin{cases} 0 & \text{if } \mathbf{z} \notin \Omega \\ 1 - \dfrac{\Delta\Omega\left(\mathbf{z}\right)}{4\pi} & \text{if } \mathbf{z} \in \Gamma \\ 1 & \text{if } \mathbf{z} \in \Omega \text{ and } \mathbf{z} \notin \Gamma \end{cases} \qquad (3.16)$$

where the integrals are defined on the sense of the Cauchy principal value.

3.1.4 Numerical Aspects

After the construction of the boundary integral equation, the next step to be evaluated, is the assembly of a linear discrete system of equations that can be solved by some type of numerical scheme. That is, the ultimate goal is to obtain a system of the type

$$\mathbf{AX} = \mathbf{B} \qquad (3.17)$$

To obtain this system, two types of approximations are necessary. First, the evaluation of integrals included in the formulation will be analyzed to subsequently, assemble the system of equations.

For the evaluation of the integrals, equation (3.15), where there is no domain integrals, is taken as an initial reference. In later sections (see for example (3.2)) the treatment of the domain integrals and its approximations within the framework of the BEM will be detailed. This chapter will focus on the classical formulation where the integrals are limited to boundary.

The first type of approximation is of geometric nature. The domain of integration Γ is divided into a series of NE elements γ_k so that

$$\Gamma \approx \sum_{k=1}^{NE} \Gamma_k \qquad (3.18)$$

At the same time, the geometry of each of these elements is approximated by a set of shape functions and discrete values of the element geometry in NA points: the so-called approximation nodes. Mathematically

$$x_k\left(\mathbf{x}\right) \approx \sum_{j=1}^{NA} \psi_k^j\left(\mathbf{x}\right) x_k^j \tag{3.19}$$

Although this formulation introduces the possibility of using different functions in each coordinate, in practice, the same type of functions are used. That is, $\psi_k^j = \psi_j$.

This technique allows to approximate complex boundaries with a high degree of accuracy using simple geometric shapes. In three-dimensional problems the most common is the use of planes triangles and quadrilaterals, but it is also extended the use of higher order polynomials. The improvement in the representation of the boundary can be achieved increasing the number of elements or increasing the order of the approximation.

The second type of approximation is of functional nature, because the integrand values are not always known, since some of its components are the unknown variables of the boundary. To deal with this aspect, the same process is used. The value of the functions defining the problem variables to be considered, is modeled by approximation functions and the discrete value (sometimes unknown) at the so-called approximation nodes. The mathematical structure obtained is identical. Assuming an isoparametric model, so that the same approximation functions which characterize the geometry are used again, we obtain

$$\phi \approx \sum_{j=1}^{NA} \psi_j \phi_j \qquad \frac{\partial \phi}{\partial \mathbf{n}} \approx \sum_{j=1}^{NA} \psi_j \left(\frac{\partial \phi}{\partial \mathbf{n}}\right)_j \tag{3.20}$$

Introducing this discretization in (3.15) we obtain

$$c\left(\mathbf{z}\right)\sum_{j=1}^{NA}\psi_j\left(\mathbf{z}\right)\phi_j + \sum_{k=1}^{NE}\int_{\Gamma_k}\sum_{j=1}^{NA}\psi_j\left(\mathbf{x}\right)\phi_j\frac{\partial w^*}{\partial \mathbf{n}}\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma_k$$

$$\quad(3.21)$$

$$=\sum_{k=1}^{NE}\int_{\Gamma_k}w^*\left(\mathbf{x};\mathbf{z}\right)\sum_{j=1}^{NA}\psi_j\left(\mathbf{x}\right)\left(\frac{\partial \phi}{\partial \mathbf{n}}\right)_j\mathrm{d}\Gamma_k$$

This discretized equation is valid for any z. Since the introduction of functional approximations of the fields, from their values at the approximation nodes, produces a linear equation with as many unknowns as approximation nodes[5], it is necessary to raise as many equations as unknowns in order to solve the system of equations.

There are mainly two techniques for building these systems of linear equations in the literature: Collocation Method [19] and Galerkin method [109].

Formally, a unified formulation can be developed if it is considered that equation (3.21) can be rewritten as

$$\mathcal{F}\left(\mathbf{z}\right)=0 \qquad (3.22)$$

And imposing that the above equation is satisfied in accordance with an integral formulation ("weak") we obtain

$$\int_{\Gamma}\varphi_i\left(\mathbf{z}\right)\mathcal{F}\left(\mathbf{z}\right)\mathrm{d}\Gamma=0 \qquad (3.23)$$

Assuming $\varphi_i\left(\mathbf{z}\right)=\delta\left(\mathbf{x}-\mathbf{z}_i\right)$, equation(3.23) is reduced to the collocation method. In this case equation (3.21) must be satisfied on a collection of k collocation nodes. If, instead, it is assumed that $\varphi_i\left(\mathbf{z}\right)=\psi_i\left(\mathbf{z}\right)$ the shape functions used for the approximations described previously are used as weighting functions to get the so called Galerkin method.

---

[5]In the case of problems with multiple variables, including the elastic problem, the number of unknowns is a multiple of the amount of approximation nodes.

Clearly, the Galerkin method presents more complexity by introducing a double integration. There are several jobs where both methods (see [121] and [9]) are compared from several perspectives. The context of this thesis is restricted to the approximation using the collocation method.

Assuming, as it was indicated previously, a formulation based on collocation, equation (3.21) can be rewritten as

$$\sum_{j=1}^{NA} h_{ij}\phi_j = \sum_{j=1}^{NA} g_{ij} \left(\frac{\partial \phi}{\partial \mathbf{n}}\right)_j \tag{3.24}$$

for a collocation node i located at $\mathbf{z}_i$, thereby identifying terms

$$h_{ij} = c\left(\mathbf{z}_i\right)\psi_j\left(\mathbf{z}_i\right) + \int_\Gamma \psi_j\left(\mathbf{x}\right)\frac{\partial w^*}{\partial \mathbf{n}}\left(\mathbf{x};\mathbf{z}_i\right)\mathrm{d}\Gamma \tag{3.25a}$$

$$g_{ij} = \int_\Gamma \psi_j\left(\mathbf{x}\right)w^*\left(\mathbf{x};\mathbf{z}_i\right)\mathrm{d}\Gamma \tag{3.25b}$$

It must be pointed out that, in general, shape functions have non-zero values only in small areas of the domain of integration. So, in practice, the domain of integration does not extend to the entire boundary. Specifically, in the case of piecewise constant functions, the domain of integration Γ for h_{ij} and g_{ij} is limited to γ_j.

After the discretization of the integral equation (3.21), the next step is the application of this discretized equation on i collocation nodes of the boundary. In this way, a determined (or overdetermined) algebraic system of equations that solves the problem is generated. This system can be written, in matrix form, as follows

$$\mathbf{H}\phi = \mathbf{G}\frac{\partial \phi}{\partial \mathbf{n}} \tag{3.26}$$

Once the system of equations is built, the introduction of the boundary conditions of the problem allow to rearrange the system and obtain a system as (3.17), where all the unknowns are gathered in the vector $\mathbf{X}$.

This system can be solved using direct methods such as LU decomposition [21] or iterative schemes as GMRES [101], to obtain $\phi\left(\mathbf{x}\right)$ and $\dfrac{\partial \phi}{\partial \mathbf{n}}\left(\mathbf{x}\right)$ on the boundary.

Following the resolution of the system, additional values on the boundary or inside the domain can be calculated from equation (3.21) with the appropriate $c\left(\mathbf{z}\right)$ function to each case. Notice that, at the post-processing phase, since the unknowns of the boundary have already been obtained, all values of the integrands are known, and the integral equations can be calculated directly for the desired values. Similarly, differentiated variables associated with the problem can be calculated by differentiating equation (3.15).

3.2 Dual Reciprocity Method

The Dual Reciprocity Boundary Element Method (DRBEM acronym in English) was introduced by *Nardini y Brebbia* [82] as an extension of the Boundary Element Method. It can treat volume integrals, that appear in the classical formulation, maintaining the boundary character of the method.

Although its main application, both historically and currently, is the study of dynamic processes (see for example [112] o [32]), DRBEM has been successfully used to deal with problems with non-homogeneous materials (see for example [111] o [76]).

As a starting point, equation (3.1), which is repeated for clarity of presentation is used.

$$\nabla \cdot \left(\mathbf{K}\left(\mathbf{x}\right)\nabla \phi\right) + b\left(\mathbf{x}\right) = 0 \qquad \boxed{3.27}$$

It shall be assumed again that the problem involves a homogeneous and isotropic material with $\mathbf{K}\left(\mathbf{x}\right) = \mathbf{I}\,k_0 = 1$ but, in this case with a non-zero independent term $b\left(\mathbf{x}\right)$.

Following an identical process as in the section focused on BEM, the construction of the integral equation, would lead to

$$c\left(\mathbf{z}\right)\phi\left(\mathbf{z}\right) + \int_{\Gamma}\left(\phi\left(\mathbf{x}\right)\frac{\partial w^{*}}{\partial\mathbf{n}}\left(\mathbf{x};\mathbf{z}\right) - w^{*}\left(\mathbf{x};\mathbf{z}\right)\frac{\partial\phi}{\partial\mathbf{n}}\left(\mathbf{x}\right)\right)\mathrm{d}\Gamma = \int_{\Omega}w^{*}\left(\mathbf{x};\mathbf{z}\right)b\left(\mathbf{x}\right)\mathrm{d}\Omega$$

(3.28)

where it is emphasized again that, unlike in equation (3.15) is not assumed that the independent term is zero.

The next step is the approximation of $b\left(\mathbf{x}\right)$ through a sum of R approximation functions. Thus

$$b\left(\mathbf{x}\right) \approx \sum_{j=1}^{R}\alpha_{j}f_{j}\left(\mathbf{x}\right)$$

(3.29)

These approximation functions f_{j} are selected in such a way that, if they are used as the independent term of the problem to solve, it is easy to obtain analytical solutions of the resulting equations.

$$\nabla^{2}\hat{\phi}_{j} + f_{j} = 0$$

(3.30)

If equation (3.28) is written using f_{j} as independent term we obtain

$$c\left(\mathbf{z}\right)\hat{\phi}_{j}\left(\mathbf{z}\right) + \int_{\Gamma}\left(\hat{\phi}_{j}\left(\mathbf{x}\right)\frac{\partial w^{*}}{\partial\mathbf{n}}\left(\mathbf{x};\mathbf{z}\right) - w^{*}\left(\mathbf{x};\mathbf{z}\right)\frac{\partial\hat{\phi}_{j}}{\partial\mathbf{n}}\left(\mathbf{x}\right)\right)\mathrm{d}\Gamma =$$

(3.31)

$$\int_{\Omega}w^{*}\left(\mathbf{x};\mathbf{z}\right)f_{j}\left(\mathbf{x}\right)\mathrm{d}\Omega$$

Introducing this expression in equation (3.28) gives

$$
c\left(\mathbf{z}\right)\phi\left(\mathbf{z}\right) + \int_{\Gamma}\left(\phi\left(\mathbf{x}\right)\frac{\partial w^{*}}{\partial \mathbf{n}}\left(\mathbf{x};\mathbf{z}\right) - w^{*}\left(\mathbf{x};\mathbf{z}\right)\frac{\partial \phi}{\partial \mathbf{n}}\left(\mathbf{x}\right)\right)\mathrm{d}\Gamma =
$$

$$
\sum_{j=1}^{R}\alpha_{j}\left[c\left(\mathbf{z}\right)\hat{\phi}_{j}\left(\mathbf{z}\right) + \int_{\Gamma}\left(\hat{\phi}_{j}\left(\mathbf{x}\right)\frac{\partial w^{*}}{\partial \mathbf{n}}\left(\mathbf{x};\mathbf{z}\right) - w^{*}\left(\mathbf{x};\mathbf{z}\right)\frac{\partial \hat{\phi}_{j}}{\partial \mathbf{n}}\left(\mathbf{x}\right)\right)\mathrm{d}\Gamma\right]
$$

$$\tag{3.32}$$

This integral equation is the basis of DRBEM formulation. From here we proceed, as in previous sections, to the discretization process for the construction of the algebraic linear system of equations.

With this in mind, we can write the discretized version of equation (3.32) in $\mathbf{z}_i$ as

$$
\sum_{k=1}^{NA}H_{ik}\phi_{k} - \sum_{k=1}^{NA}G_{ik}\left(\frac{\partial \phi}{\partial \mathbf{n}}\right)_{k} = \sum_{j}^{R}\alpha_{j}\left[\sum_{k=1}^{NA}H_{ik}\hat{\phi}_{kj} - \sum_{k=1}^{NA}G_{ik}\left(\frac{\partial \hat{\phi}}{\partial \mathbf{n}}\right)_{kj}\right] \tag{3.33}
$$

which can be written in matrix form as

$$
\mathbf{H}\phi - \mathbf{G}\frac{\partial \phi}{\partial \mathbf{n}} = \left[\mathbf{H}\hat{\phi} - \mathbf{G}\frac{\partial \hat{\phi}}{\partial \mathbf{n}}\right]\boldsymbol{\alpha} \tag{3.34}
$$

On the other hand if (3.29) is particularized in a number of evaluation points $\mathbf{x}_i \in \Omega$ we obtain

$$
b\left(\mathbf{x}_{i}\right) \approx \sum_{j=1}^{R}\alpha_{j}f_{j}\left(\mathbf{x}_{i}\right) \tag{3.35}
$$

which can be written in matrix form as

$$
\mathbf{b} = \mathbf{F}\boldsymbol{\alpha} \tag{3.36}
$$

Although the coefficients α_j can be obtained from equation (3.36) and, by introducing them in (3.34) the system of equations can be solved, the following procedure is traditionally performed.

Assuming that the approximation (3.29) is particularized in a number of points,to produce an invertible $\mathbf{F}$ matrix, we get

$$\boldsymbol{\alpha} = \mathbf{F}^{-1}\mathbf{b} \qquad (3.37)$$

If (3.37) is introduced in (3.34) we get

$$\mathbf{H}\boldsymbol{\phi} - \mathbf{G}\frac{\partial \phi}{\partial \mathbf{n}} = \left[\mathbf{H}\hat{\boldsymbol{\phi}} - \mathbf{G}\frac{\partial \hat{\phi}}{\partial \mathbf{n}}\right]\mathbf{F}^{-1}\mathbf{b} = \mathbf{S}\mathbf{b} \qquad (3.38)$$

This linear system of equations allows to solve the original problem, including the addition of the independent term, maintaining the boundary character of the method. It is easy to check that if the choice of the approximation functions is restricted to solutions of the original differential operator the new integrals appearing in the DRM methodology are confined to the boundary.

3.3 Approximation and Interpolation of functions

One aspect of the dual reciprocity methods which has not been analyzed in the previous section, is the nature of the approximation functions f_j. This aspect has its roots in the general problem of function approximation, which exceeds the objectives of this work, and for which there are abundant references in literature (see, for example, [20]).

First of all,the field of study is restricted to Dual Reciprocity Methods related works (see [55], [91], [20], [92] or [83]), which means that the approximation functions must have analytic solution (or with very low computational cost) of the differential operator of the problem. In addition, the mathematical

treatment necessary for the implementation of the approximation requires the use of relatively simple explicit analytical expressions.

A preliminary study of these methods almost entirely excludes the so-called local methods. These methods base their approximation scheme on sets of functions $f(x)$, depending on a small number of near values, to approximate data. This means that the definition of the approximation function is changing throughout the domain, and in many cases it is not even continuous. These two factors discard these families of approximations functions as viable solutions in the DRM.

Similarly, approximation schemes using piecewise or spline functions, may formally be desirable, but involve significant complexity in implementation and its use in the DRM schemes are also discarded. In general, we can restrict the range of approximation functions appropriate for use in the DRM approach to three main groups.

- Radial basis functions (RBF acronym in English)

- Global Functions

- Radial basis functions augmented with global functions

These categories may have subdivisions, for example, depending on the types of functions or the existence of parameters whose value is selectable.

In this general classification, it can be noted that the definition of the groups allows, naturally, to build a unified formulation that includes the three groups. This can be done using coefficients that vanish or produce non-zero values depending on the type of approximation chosen. Of the three above groups, the use of the second one (global functions) is clearly a minority. The use of global functions is focused on improving radial basis functions approximation (the third group). Although there are more possibilities of approximation functions as polynomial functions, radial basis functions have advantages over these, in terms of existence of interpolating and convergence. This is mainly due to its radial symmetry, smoothness, and and certain properties of their Fourier transform (see [20] for details).

3.3. APPROXIMATION AND INTERPOLATION OF FUNCTIONS

The approximation by radial basis functions is a general method of approximation of multivariate functions by sets of radial basis functions, ie, based on distances to nodes. More formally, we seek to approximate functions of the type $h : \mathbb{R}^n \to \mathbb{R}$ using approximants of the type

$$h\left(\mathbf{x}\right) \approx \sum_{j=1}^{M} \alpha_j f_j\left(\|\mathbf{x} - \mathbf{x}_j\|\right) + \sum_{k=1}^{N} \beta_k g_k\left(\mathbf{x}\right) \qquad (3.39)$$

whose α_j and β_k coefficients can be determined by equations

$$\sum_{j=1}^{M} \alpha_j f_j\left(\|\mathbf{x}_i - \mathbf{x}_j\|\right) + \sum_{k=1}^{N} \beta_k g_k\left(\mathbf{x}_i\right) = h(\mathbf{x}_i) \qquad i = 1, \dots M$$

$$\sum_{j=1}^{M} \alpha_j g_i\left(\mathbf{x}_j\right) = 0 \qquad i = 1, \dots N$$

$$(3.40)$$

In the above equations we have used the generalized formulation covering the three groups of approximation functions depending on whether the terms β_k vanish (radial basis functions), the terms α_j vanish (global functions) or both have non-zero values (radial basis functions augmented with global functions).

The approximation of functions, by means of radial basis functions, have been widely used in Boundary Element formulations, mainly coupled with DRM algorithms (see, for example [93], [43], [33] o [113]).

The following list includes, without being exhaustive, some of the radial basis functions typically used in BEM methods

- Euclidian distance $f(r) = c + r$

- Gauss $f(r) = e^{-cr^2}$

- Multiquadrics $f(r) = \sqrt{c + r^2}$

- Inverse Multiquadrics $f(r) = \dfrac{1}{\sqrt{c + r^2}}$

- Thin plate Spline $f(r) = r^2 \log r$

- Radial basis functions families with support

  - Wendland family e.g. $f(r) = \left(1 - \dfrac{r}{s}\right)^4 \left(1 + \dfrac{4r}{s}\right)$ for $r < s$

  - Buhmann family e.g. $f(r) = 2\left(\dfrac{r}{s}\right)^4 \log \dfrac{r}{s} - \dfrac{7}{2}\left(\dfrac{r}{s}\right)^4 + \dfrac{16}{3}\left(\dfrac{r}{s}\right)^3 - 2\left(\dfrac{r}{s}\right)^2 + \dfrac{1}{6}$ for $r < s$

Versions of these functions improved with global functions (referred with the term "augmented") are also commonly used. In general, these additions, and their associated compatibility equations (3.40), are designed to improve system stability in the direct approximation problem (see for example [56]).

In the more typical case, with polynomial terms up to order one, the compatibility equations in three-dimensional problem are

$$\sum_{j=1}^{M} \alpha_j = \sum_{j=1}^{M} \alpha_j x_1\left(\mathbf{x}_j\right) = \sum_{j=1}^{M} \alpha_j x_2\left(\mathbf{x}_j\right) = \sum_{j=1}^{M} \alpha_j x_3\left(\mathbf{x}_j\right) = 0 \qquad \boxed{3.41}$$

The analysis of convergence, stability, accuracy, selection of parameters optimal values... of these approximation functions (within the framework of Boundary Element Methods) are aspects whose investigation remains open and exceeds the objectives of this work (see for example [83], [45] o [23] among others).

3.4 Analog Equation Method

The Analog Equation Method (AEM acronym in English) was introduced by Katsikadelis in 1994 [58] as a method derived from BEM, capable of treating domain integrals without discretizing the domain (as in the Dual Reciprocity Methods). It has been applied successfully to linear or nonlinear

problems [63] and to problems involving inhomogeneous materials [60] among others.

The basis of the methodology is, in essence, very simple. The key idea is to replace the differential operator of the original problem, whose fundamental solution has a high computational cost or simply is not available in the literature, for a linear differential operator with known fundamental solution, by using the analog equation.

Let consider a problem in a domain $\Omega \subseteq \mathbb{R}^3$ enclosed by a boundary Γ and governed by the following boundary value problem

$$\mathcal{L}(u) + b(\mathbf{x}) = 0 \quad \text{in } \mathbf{x} \in \Omega \tag{3.42a}$$

$$\lambda(\mathbf{x}) u + \beta(\mathbf{x}) q = \gamma(\mathbf{x}) \quad \text{on } \mathbf{x} \in \Gamma \tag{3.42b}$$

where $\mathcal{L}$ is a linear second order differential operator, $\mathbf{b}(\mathbf{x})$ is the known domain loading source, $\lambda(\mathbf{x}), \beta(\mathbf{x})$ and $\gamma(\mathbf{x})$ are functions defined on the boundary Γ and $q = \dfrac{\partial u}{\partial \mathbf{n}}$ is the flux of u. Without loss of generality we are dealing with a scalar field $\mathbf{u} = \mathbf{u}(\mathbf{x})$ defined in a two or three-dimensional space (although the Analog Equation Method can deal with vectorial problems with more variables, such as elasticity [86], poro-elasticity [85], etc...)

It must be pointed out as well that with this definition, the operator $\mathcal{L}$ can represent a wide variety of physical problems involving homogeneous or inhomogeneous materials [58], plane elastostatics [62]...

The next step is to select an auxiliary operator $\overset{o}{\mathcal{L}}$, called analog operator, with known fundamental solution. For the sake of simplicity, but without loss of generality, the Laplace operator is chosen as analog operator. Thus, if the Laplace operator is applied to the sought solution of the problem defined by equations (3.42a) and (3.42b), the problem is transformed into

$$\nabla^2 u + \hat{b}(\mathbf{x}) = 0 \quad \text{in } \mathbf{x} \in \Omega \tag{3.43a}$$

$$\lambda(\mathbf{x}) u + \beta(\mathbf{x}) q = \gamma(\mathbf{x}) \quad \text{on } \mathbf{x} \in \Gamma \tag{3.43b}$$

where $\hat{b}$ represents an unknown fictitious source . In the original methodology, introduced by Katsikadelis [58], the solution of the problem defined by (3.43a) and (3.43b) is decomposed into the sum of the solution of the homogeneous problem u_h and a particular solution u_p. Thus

$$u\left(\mathbf{x}\right) = u_h\left(\mathbf{x}\right) + u_p\left(\mathbf{x}\right) \qquad (3.44)$$

The next step is to approximate the unknown fictitious source by a sum of R approximation functions

$$\hat{b}\left(\mathbf{x}\right) \approx \sum_{j=1}^{R} \alpha_j f_j\left(\mathbf{x}\right) \qquad (3.45)$$

If the f_j functions are selected, so that analytical solutions of the analog operator with this set of functions as source terms is available we can write

$$\nabla^2 \hat{u}_j + f_j\left(\mathbf{x}\right) = 0 \qquad (3.46)$$

Although not formally required, noticed that the functions f_j used in these methodologies are, mostly, radial basis functions. With this in mind, the particular solution u_p can be approximated by

$$u_p\left(\mathbf{x}\right) \approx \sum_{j=1}^{R} \alpha_j \hat{u}_j\left(\mathbf{x}\right) \qquad (3.47)$$

The homogeneous solution is obtained using standard BEM methodology so the Boundary Integral Equation is constructed to obtain

$$c\left(\mathbf{z}\right) u_h\left(\mathbf{z}\right) = -\int_{\Gamma} \left(u_h\left(\mathbf{x}\right) \frac{\partial w^*}{\partial \mathbf{n}}\left(\mathbf{x};\mathbf{z}\right) - w^*\left(\mathbf{x};\mathbf{z}\right) q_h\left(\mathbf{x}\right) \right) \mathrm{d}\Gamma \qquad (3.48)$$

which can be written in matrix form as

$$\mathbf{H}\mathbf{U_h} - \mathbf{G}\mathbf{Q_h} = 0 \qquad \boxed{3.49}$$

If this equation is particularized at an interior point of the domain (in the Figure 3.3 is represented a distribution of internal nodes for a two-dimensional domain), the function $c(\mathbf{z}) = 1$, thus taking into account (3.44) and (3.47) we conclude that

$$u(\mathbf{z}) = -\int_{\Gamma} \left(u_h(\mathbf{x}) \frac{\partial w^*}{\partial \mathbf{n}}(\mathbf{x}; \mathbf{z}) - w^*(\mathbf{x}; \mathbf{z}) q_h(\mathbf{x}) \right) d\Gamma + \sum_{j=1}^{R} \alpha_j \hat{u}_j(\mathbf{z}) \qquad \boxed{3.50}$$
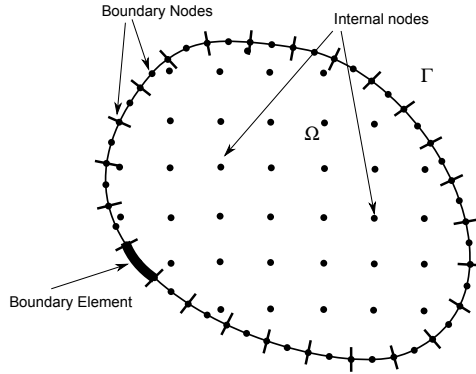
Figure 3.3: Two-dimensional discretization of a boundary and distribution of internal nodes

By applying the differential operator of the original problem $\mathcal{L}$ to equation (3.50) we get

$$b(\mathbf{z}) = \int_{\Gamma} \left(u_h(\mathbf{x}) \mathcal{L} \left[\frac{\partial w^*}{\partial \mathbf{n}}(\mathbf{x}; \mathbf{z}) \right] - \mathcal{L}[w^*(\mathbf{x}; \mathbf{z})] q_h(\mathbf{x}) \right) d\Gamma - \sum_{j=1}^{R} \alpha_j \mathcal{L}[\hat{u}_j(\mathbf{z})]$$

$$\boxed{3.51}$$

where the derivatives are taken with respect to coordinate z. If this equation is applied to all internal points, as shown in Figure 3.3, and the resulting equation is written in matrix form we obtain

$$\mathbf{K}\mathbf{U_h} - \mathbf{L}\mathbf{Q_h} - \hat{\mathbf{B}}\alpha = \mathbf{B} \qquad (3.52)$$

where

$$\mathbf{K} = \mathcal{L}\left[\mathbf{H}\right] \qquad \mathbf{L} = \mathcal{L}\left[\mathbf{G}\right] \qquad \hat{\mathbf{B}} = \mathcal{L}\left[\hat{\mathbf{U}}\right] \qquad (3.53)$$

It must be pointed out that equation (3.51) is applied to an internal point where $c(\mathbf{z})$ is constant and therefore its derivative is zero. Consequently, the coefficients of the matrix $\mathbf{H}$ which are affected by the derivative can be written as

$$\bar{h}_{ij} = h_{ij} - c\left(\mathbf{z}_i\right)\psi_j = \int_\Gamma \psi_j\left(\mathbf{x}\right)\frac{\partial w^*}{\partial \mathbf{n}}\left(\mathbf{x};\mathbf{z}_i\right)\mathrm{d}\Gamma \qquad (3.54)$$

Taking into account the definition of u_p as it is indicated in (3.45), equation (3.44) can be rewritten as

$$u\left(\mathbf{x}\right) \approx u_h\left(\mathbf{x}\right) + \sum_{j=1}^{R}\alpha_j \hat{u}_j\left(\mathbf{x}\right) \qquad (3.55)$$

and, consequently, the associated flux $q = \dfrac{\partial u}{\partial \mathbf{n}}$

$$q\left(\mathbf{x}\right) \approx q_h\left(\mathbf{x}\right) + \sum_{j=1}^{R}\alpha_j \hat{q}_j\left(\mathbf{x}\right) \qquad (3.56)$$

If these two representations are introduced into the definition of the boundary condition as is defined in (3.43b) we conclude that

$$\lambda\left(\mathbf{x}\right)\left[u_h\left(\mathbf{x}\right)+\sum_{j=1}^{R}\alpha_j\hat{u}_j\left(\mathbf{x}\right)\right]+\beta\left(\mathbf{x}\right)\left[q_h\left(\mathbf{x}\right)+\sum_{j=1}^{R}\alpha_j\hat{q}_j\left(\mathbf{x}\right)\right]=\gamma\left(\mathbf{x}\right)\quad\text{en }\mathbf{x}\in\Gamma$$

$$(3.57)$$

which can be written in matrix form as

$$\mathbf{\Lambda}\left[\mathbf{U_h}+\hat{\mathbf{U}}\boldsymbol{\alpha}\right]+\mathbf{B}\left[\mathbf{Q_h}+\hat{\mathbf{Q}}\boldsymbol{\alpha}\right]=\boldsymbol{\gamma} \qquad (3.58)$$

where the diagonal matrices $\boldsymbol{\lambda}$ and $\mathbf{B}$ have been introduced. These matrices contain on its diagonal the value at collocation nodes of the functions λ and β.

Equations (3.49), (3.52) and (3.58) constitute a system of equations which can be solved to obtain the boundary quantities q_h and u_h, as well as the α coefficients that define u_p. Using these values the solution of the original problem defined by u and q can be calculated.

As in the previous section, it is possible to manipulate equation (3.58) to obtain an expression that relates $\boldsymbol{\alpha}$ with $\mathbf{U_h}$ and $\mathbf{Q_h}$, to obtain, in a first step, the values $\mathbf{U_h}, \mathbf{Q_h}$ of the homogeneous solution and subsequently the α coefficients that define the particular solution.

Similarly as described in section 3.1, after obtaining the values of the homogeneous solution on the boundary and the values of the α coefficients that define the particular solution , equation (3.50) can be used for obtaining the values of the solution at any point of the domain.

3.4.1 Analog Equation Method combined with Dual Reciprocity Method

In 2005, Katsikadelis [60] introduced a variation of the method combining the Analog equation Method with the Dual Reciprocity Method. Starting

from the same problem raised in the previous section, that it is repeated for sake of clarity

$$\mathcal{L}(u) + b\left(\mathbf{x}\right) = 0 \quad \text{en } \mathbf{x} \in \Omega \tag{3.59a}$$

$$\alpha\left(\mathbf{x}\right) u + \beta\left(\mathbf{x}\right) q = \gamma\left(\mathbf{x}\right) \quad \text{en } \mathbf{x} \in \Gamma \tag{3.59b}$$

if the analog operator is applied, we get again

$$\nabla^2 u + \hat{b}\left(\mathbf{x}\right) = 0 \quad \text{in } \mathbf{x} \in \Omega \tag{3.60a}$$

$$\alpha\left(\mathbf{x}\right) u + \beta\left(\mathbf{x}\right) q = \gamma\left(\mathbf{x}\right) \quad \text{on } \mathbf{x} \in \Gamma \tag{3.60b}$$

Using the same approximation of the unknown fictitious source term we get

$$\hat{b}\left(\mathbf{x}\right) \approx \sum_{j=1}^{R} \alpha_j f_j\left(\mathbf{x}\right) \tag{3.61}$$

In this case, the boundary integral equation is used directly, following the standard BEM process results in an equation similar to (3.28). Therefore we obtain

$$c\left(\mathbf{z}\right) u\left(\mathbf{z}\right) + \int_{\Gamma} \left(u\left(\mathbf{x}\right) \frac{\partial w^*}{\partial \mathbf{n}}\left(\mathbf{x}; \mathbf{z}\right) - w^*\left(\mathbf{x}; \mathbf{z}\right) q\left(\mathbf{x}\right) \right) \mathrm{d}\Gamma = \int_{\Omega} w^*\left(\mathbf{x}; \mathbf{z}\right) \hat{b}\left(\mathbf{x}\right) \mathrm{d}\Omega \tag{3.62}$$

and by means of the Dual Reciprocity Method

$$c\left(\mathbf{z}\right) u\left(\mathbf{z}\right) + \int_{\Gamma} \left(u\left(\mathbf{x}\right) \frac{\partial w^*}{\partial \mathbf{n}}\left(\mathbf{x}; \mathbf{z}\right) - w^*\left(\mathbf{x}; \mathbf{z}\right) q\left(\mathbf{x}\right) \right) \mathrm{d}\Gamma =$$

$$\sum_{j=1}^{R} \alpha_j \left[c\left(\mathbf{z}\right) \hat{u}_j\left(\mathbf{z}\right) + \int_{\Gamma} \left(\hat{u}_j\left(\mathbf{x}\right) \frac{\partial w^*}{\partial \mathbf{n}}\left(\mathbf{x}; \mathbf{z}\right) - w^*\left(\mathbf{x}; \mathbf{z}\right) \hat{q}_j\left(\mathbf{x}\right) \right) \mathrm{d}\Gamma \right] \tag{3.63}$$

and using again matrix notation

$$\mathbf{HU} - \mathbf{GQ} = \left[\mathbf{H\hat{U}} - \mathbf{G\hat{Q}}\right]\boldsymbol{\alpha} \qquad \text{(3.64)}$$

Following the same process as in the previous section, equation (3.63) is particularized at an internal point of the domain. Thereby applying the differential operator with respect to coordinate z we obtain

$$-b\left(\mathbf{z}\right) + \int_{\Gamma}\left(u\left(\mathbf{x}\right)\mathcal{L}\left[\frac{\partial w^*}{\partial \mathbf{n}}\left(\mathbf{x};\mathbf{z}\right)\right] - \mathcal{L}\left[w^*\left(\mathbf{x};\mathbf{z}\right)\right]q\left(\mathbf{x}\right)\right)d\Gamma =$$

$$\text{(3.65)}$$

$$\sum_{j=1}^{R}\alpha_j\mathcal{L}\left[\hat{u}_j\left(\mathbf{z}\right)\right] + \int_{\Gamma}\left(\hat{u}_j\left(\mathbf{x}\right)\mathcal{L}\left[\frac{\partial w^*}{\partial \mathbf{n}}\left(\mathbf{x};\mathbf{z}\right)\right] - \mathcal{L}\left[w^*\left(\mathbf{x};\mathbf{z}\right)\right]\hat{q}_j\left(\mathbf{x}\right)\right)d\Gamma$$

which can be written in matrix form as

$$\mathbf{KU} - \mathbf{LQ} = \left[\mathbf{\hat{B}} + \mathbf{K\hat{U}} - \mathbf{L\hat{Q}}\right]\boldsymbol{\alpha} + \mathbf{B} \qquad \text{(3.66)}$$

where once again

$$\mathbf{K} = \mathcal{L}\left[\mathbf{H}\right] \qquad \mathbf{L} = \mathcal{L}\left[\mathbf{G}\right] \qquad \mathbf{\hat{B}} = \mathcal{L}\left[\mathbf{\hat{U}}\right] \qquad \text{(3.67)}$$

In this case variables u and q are used, so the boundary conditions can be applied directly in equations (3.64) y (3.66), in order to construct the linear system of equations which allows to solve the problem.

Part II

Contributions

4

General Analog Equation

The Analog Equation Method was introduced by Katsikadelis in 1994 [58]. The main features of this formulation have been presented briefly in section (3.4). This methodology is able to solve general problems using the Laplace operator as analog operator. Notice that the original form of this formulation implied that the boundary conditions of the problems must be linear combinations of the variables involved and their fluxes.

Later, in 2005 [60], a variation of the original formulation was introduced. That work conjugated the analog equation method with Dual Reciprocity Methods (see (3.4.1)), keeping the same limitation on the definition of boundary conditions. Although, in subsequent works [62], the elastostatics problem with Neumann (stress) boundary conditions was analyzed, the treatment of these boundary conditions is not performed through the use of integral equations. Instead, vector decomposition and numerical approximation, by finite differences, of the tangential derivatives is used.

To illustrate this, one can assume a general elastic problem in a two dimensional space with stress conditions. The operator associated with stress is defined as

$$t_i\left(\mathbf{x}\right) = C_{ijkl}\left(\mathbf{x}\right)\left[\frac{\partial u_k\left(\mathbf{x}\right)}{\partial x_l}\right]n_j\left(\mathbf{x}\right) \qquad (4.1)$$

where, respectively, u_i and t_i are the components of displacement and stress. Manipulating the equation, we can rewrite this expression as

$$
\begin{aligned}
t_i = C_{ijkl}\left(\mathbf{x}\right) & \left[\frac{\partial u_k\left(\mathbf{x}\right)}{\partial \mathbf{n}}\cos(\mathbf{x},\mathbf{n}) + \frac{\partial u_k\left(\mathbf{x}\right)}{\partial \mathbf{t}}\cos(\mathbf{x},\mathbf{t})\right] n_j\left(\mathbf{x}\right) = \\
C_{ijkl}\left(\mathbf{x}\right) & \left[q_k\left(\mathbf{x}\right)\cos(\mathbf{x},\mathbf{n}) + \frac{\partial u_k\left(\mathbf{x}\right)}{\partial \mathbf{t}}\cos(\mathbf{x},\mathbf{t})\right] n_j\left(\mathbf{x}\right)
\end{aligned}
\tag{4.2}
$$

where, in addition to the normal derivative, an unknown tangential derivative appears. This tangential derivative is approximated by numerical differentiation. Finally, it can be noted that the practical implementation of these algorithms has been conducted only in two-dimensional problems.

In this chapter a set of integral equations is formulated for a generic analog operator, able to solve, after discretization, boundary value problems for any linear differential operator and generic boundary conditions.

4.1 Problem statement

Let consider a problem defined in a domain Ω whose boundary is Γ, governed by the following general boundary value problem,

$$
\mathcal{L}_i(\mathbf{u}) + b_i\left(\mathbf{x}\right) = 0 \quad \text{in } \mathbf{x} \in \Omega \quad i = 1, d \tag{4.3a}
$$

$$
\mathcal{G}_i(\mathbf{u}) + \gamma_i\left(\mathbf{x}\right) = 0 \quad \text{on } \mathbf{x} \in \Gamma \quad i = 1, d \tag{4.3b}
$$

This formulation can be applied, for both scalar and vector problems of dimension d, as well as in two and three-dimensional spaces.

$\mathcal{L}_i$ and $\mathcal{G}_i$ are, respectively, second and first order linear differential operators applied to the field components $\mathbf{u}$. The functions $b_i\left(\mathbf{x}\right)$ and $\gamma_i\left(\mathbf{x}\right)$ are the independent terms of equations (4.3a) and (4.3b), which are defined in Ω and Γ respectively.

The definition of these operators is quite general and covers a large number of problems of interest in engineering, including problems involving homogeneous or inhomogeneous materials. Although it is possible to extend this methodology to include dynamic problems (see e.g. [59] or [88]), such extension is beyond the scope of this work.

In the following sections of this chapter, three general sets of integral equations are deduced. The system of equations comprising the three sets allows to solve the general problem formulated above.

- u-BIE[1] is the set of basic integral equations of BEM.

- q-BIE is the set of equations consisting of sums of u-BIE and its derivatives, up to first order, associated with the boundary conditions.

- b-BIE is the set of equations consisting of sums of u-BIE and its derivatives, up to second order, associated with differential operator that governs the problem.

4.2 General u-BIE equation

Following the methodology introduced in the previous chapter, we assume the existence of the solution $\mathbf{u}(\mathbf{x})$. Then, The analog operator is applied to each component of $\mathbf{u}(\mathbf{x})$. Generally, we are dealing with a vector operator of dimension d.

With this in mind, the system of equations consisting of (4.3a) and (4.3b) is transformed into

$$\overset{o}{\mathcal{L}}_i(\mathbf{u}) + \hat{b}_i(\mathbf{x}) = 0 \quad \text{in } \mathbf{x} \in \Omega \quad i = 1, d \qquad \boxed{4.4\text{a}}$$

$$\mathcal{G}_i(\mathbf{u}) + \gamma_i(\mathbf{x}) = 0 \quad \text{on } \mathbf{x} \in \Gamma \quad i = 1, d \qquad \boxed{4.4\text{b}}$$

---

[1]BIE - Boundary Integral Equation.

where, again, an unknown independent term $\hat{b}_i$ appears. Without loss of generality, it is assumed that the fundamental solution of the analog operator is available in analytic form.

$$\overset{o}{\mathcal{L}}(\mathbf{u}) + \delta\left(\mathbf{x} - \mathbf{z}\right)\mathbf{e}_m = 0 \Longrightarrow \overset{o}{U}_i^m\left(\mathbf{x}; \mathbf{z}\right), \overset{o}{Q}_i^m\left(\mathbf{x}; \mathbf{z}\right) \qquad \boxed{4.5}$$

where $\overset{o}{U}_i^m$ represents the "displacement" at any point in the i direction produced by a source applied in the m direction. $\overset{o}{Q}_i^m$ is a derivative of $\overset{o}{U}_i^m$ which specific formulation depends on the selected analog operator. In the case of Laplace operator, it represents a flux, this is

$$\overset{o}{Q}_i^m = \frac{\partial \overset{o}{U}_i^m}{\partial \mathbf{n}} \qquad \boxed{4.6}$$

As it has been indicated, the formulation is discussed in a general way. This section does not examine in depth the structure of this generic fundamental solution, and simply assumed that it is known.

Following a similar procedure as described in the previous chapter, a u-BIE integral equation can be constructed. The construction of this equation, which is formally identical to that of any vector problem as elastic, can be found in many textbooks (see e.g. [19]) and its deduction is omitted in this work. So starting from[2]

$$c_i^m\left(\mathbf{z}\right)u_i\left(\mathbf{z}\right) + \int_\Gamma u_i\left(\mathbf{x}\right)\overset{o}{Q}_i^m\left(\mathbf{x}; \mathbf{z}\right)\mathrm{d}\Gamma = \int_\Gamma q_i\left(\mathbf{x}\right)\overset{o}{U}_i^m\left(\mathbf{x}; \mathbf{z}\right)\mathrm{d}\Gamma + \int_\Omega \hat{b}_i\left(\mathbf{x}\right)\overset{o}{U}_i^m\left(\mathbf{x}; \mathbf{z}\right)\mathrm{d}\Omega \qquad \boxed{4.7}$$

$q_i\left(\mathbf{x}\right)$ is a variable calculated by applying a differential operator, which depends on the analog operator, on the vector of variables $\mathbf{u}\left(\mathbf{x}\right)$. For

---

[2]Notice that, as indicated above, in these equations, the integrals must be understood in the sense of Cauchy principal value and the free terms are grouped in $c_i^m\left(\mathbf{z}\right)$.

example, if elastic operator is used as analog operator, stress appears in the integral equations, which means that $q_i \longrightarrow t_i$

In this equation, unlike in the previous chapter, generally, a new set of unknowns $\hat{b}_i$ appears. These new unknowns are the variables q_i. These new unknowns appear because the boundary conditions of the original operator, are not expressed with the same variables that arise in the integral equation associated with the analog operator (4.7). For example, if the original operator was the elastic one, the boundary conditions consist of combinations of displacement and stress. Respectively, if the analog operator is the Laplace operator, displacements and fluxes appear in the integral equations.

The next step is the application of the Dual Reciprocity Method to transform the volume integral (4.7) into a sum of boundary integrals. In order to achieve this transformation the unknown independent term $\hat{b}_i$ is approximated by a set of R_i approximation functions f_{j_i}, so

$$\hat{b}_i\left(\mathbf{x}\right) \approx \sum_{j_i}^{R_i} \alpha_{j_i} f_{j_i}\left(\mathbf{x}\right) \tag{4.8}$$

The subscript j_i is introduced because, formally, the approximation functions, both in number and definition may be different for each index $\hat{b}_i$. Although there is no formal impediment, since the field to approximate is a priori unknown, a simplification is introduced. It is assumed that the approximation functions for each component of $\mathbf{b}$ have the same mathematical definition[3] and the same number of elements. Taking this into account, equation (4.8) is rewritten as

$$\hat{b}_i\left(\mathbf{x}\right) \approx \sum_{j}^{R} \alpha_j^i f_j\left(\mathbf{x}\right) \tag{4.9}$$

---

[3]The use of different types of approximation functions would complicate the implementation of the algorithm significantly, and it would require the definition of different sets of functions with different base point positions in each coordinate.

These f_j approximation functions are chosen so that the analytical solution of the analog operator with this set of functions as source terms is available, so

$$\overset{o}{\mathcal{L}}(\mathbf{u}) + \delta_{kj}\delta_{li}f_k(\mathbf{x})\,\mathbf{e}_l = 0 \quad \Longrightarrow \quad \hat{\mathbf{u}}_j^i(\mathbf{x}),\hat{\mathbf{q}}_j^i(\mathbf{x}) \qquad \boxed{4.10}$$

(4.7) Introducing (4.10)in equation (4.7), and by means of the Dual Reciprocity Method (detailed in the previous chapter) the domain integral is transferred to the boundary to result in

$$\boxed{\begin{aligned} &c_i^m(\mathbf{z})\,u_i(\mathbf{z}) + \int_\Gamma u_i(\mathbf{x})\,\overset{0}{Q}_i^m(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma - \int_\Gamma q_i(\mathbf{x})\,\overset{0}{U}_i^m(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma = \\ &\sum_j^R \alpha_j^p\left[c_i^m(\mathbf{z})\,\hat{u}_{ji}^p(\mathbf{z}) + \int_\Gamma \hat{u}_{ji}^p(\mathbf{x})\,\overset{0}{Q}_i^m(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma - \int_\Gamma \hat{q}_{ji}^p(\mathbf{x})\,\overset{0}{U}_i^m(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma \right] \end{aligned}} \qquad \boxed{4.11}$$

Where $\hat{u}_{ji}^p$ and $\hat{q}_{ji}^p$ are each of the components of the vectors $\hat{\mathbf{u}}_j^p$ and $\hat{\mathbf{u}}_j^p$.

4.3 General b-BIE Equation

If equation (4.11) is particularized for a point $\mathbf{z}$ inside the domain, the function $c_i^m(\mathbf{z})$, which includes the free terms of the integral equation is reduced to δ_{mi}. Taking that into account the equation is transformed into

$$\begin{aligned} &u_m(\mathbf{z}) + \int_\Gamma u_i(\mathbf{x})\,\overset{0}{Q}_i^m(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma - \int_\Gamma q_i(\mathbf{x})\,\overset{0}{U}_i^m(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma = \\ &\sum_j^R \alpha_j^p\left[\hat{u}_{jm}^p(\mathbf{z}) + \int_\Gamma \hat{u}_{ji}^p(\mathbf{x})\,\overset{0}{Q}_i^m(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma - \int_\Gamma \hat{q}_{ji}^p(\mathbf{x})\,\overset{0}{U}_i^m(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma \right] \end{aligned} \qquad \boxed{4.12}$$

Rewriting the equation, so the variables are grouped in a vector according to the m index, i.e.

$$u_m \Longrightarrow \mathbf{u} \qquad \hat{u}^p_{jm} \Longrightarrow \hat{\mathbf{u}}^p_j$$
$$\overset{0}{Q}^m_i \Longrightarrow \overset{0}{\mathbf{Q}}_i \qquad \overset{0}{U}^m_i \Longrightarrow \overset{0}{\mathbf{U}}_i$$

(4.13)

we obtain[4]

$$\mathbf{u}\left(\mathbf{z}\right) + \int_\Gamma u_i\left(\mathbf{x}\right) \overset{0}{\mathbf{Q}}_i\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma - \int_\Gamma q_i\left(\mathbf{x}\right) \overset{0}{\mathbf{U}}_i\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma =$$
$$\sum_j^R \alpha^p_j \left[\hat{\mathbf{u}}^p_j\left(\mathbf{z}\right) + \int_\Gamma \hat{u}^p_{ji}\left(\mathbf{x}\right) \overset{0}{\mathbf{Q}}_i\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma - \int_\Gamma \hat{q}^p_{ji}\left(\mathbf{x}\right) \overset{0}{\mathbf{U}}_i\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma \right]$$

(4.14)

If the differential operator $\boldsymbol{\mathcal{L}}$ of the original problem is applied to equation (4.12) with respect to $\mathbf{z}$ we obtain

$$-b_m\left(\mathbf{z}\right) + \int_\Gamma u_i\left(\mathbf{x}\right) \overset{0}{K}^m_i\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma - \int_\Gamma q_i\left(\mathbf{x}\right) \overset{0}{L}^m_i\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma =$$
$$\sum_j^R \alpha^p_j \left[\hat{\beta}^p_{jm}\left(\mathbf{z}\right) + \int_\Gamma \hat{u}^p_{ji}\left(\mathbf{x}\right) \overset{0}{K}^m_i\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma - \int_\Gamma \hat{q}^p_{ji}\left(\mathbf{x}\right) \overset{0}{L}^m_i\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma \right]$$

(4.15)

where

$$\hat{\beta}^p_{jm} = \mathcal{L}_m(\hat{\mathbf{u}}^p_j) \qquad \overset{0}{K}^m_i = \mathcal{L}_m(\overset{0}{\mathbf{Q}}_i) \qquad \overset{0}{L}^m_i = \mathcal{L}_m(\overset{0}{\mathbf{U}}_i)$$

(4.16)

It is important to remark that equation (4.15) is only valid for a point $\mathbf{z}$ located inside the domain. In case that $\mathbf{z} \in \Gamma$ the equation is singular and it requires a detailed analysis of its behavior close to the singularity. This study allows to improve the interpolation functions $\hat{b}_i\left(\mathbf{x}\right)$ by covering completely its domain of definition. Given the complexity of this study and that errors obtained in the direct problems, without boundary nodes, are small, this aspect is not addressed in this work.

---

[4]Note that, in this case, the integral equation has no singular terms because $\mathbf{z}$ is an interior point.

4.4 General q-BIE Equation

In order to obtain the integral version of the boundary conditions (represented by equation (4.3b)) a similar procedure as depicted in the previous section is employed. It must be pointed out that this procedure, although performed generically, requires some clarification, because it involves singular functions.

As it was previously stated, in general, the operator $\mathcal{G}$ is a linear combination of $\mathbf{u}$ and its first-order derivatives. Then, without loss of generality we can define

$$\mathcal{G}_m(\mathbf{u}) = f_k^m(\mathbf{z})\, u_k + g_{kl}^m(\mathbf{z})\, u_{k,l} \qquad \mathbf{z} \in \Gamma \tag{4.17}$$

To construct the integral version of equation 4.17, integral expressions for $u_k(\mathbf{z})$ and $u_{k,l}(\mathbf{z})$ must be obtained.

For the terms associated with $u_k(\mathbf{z})$, equation (4.11) is taken as starting point. If we call $\mathbf{C}$ to the matrix defined by the coefficients $c_i^l(\mathbf{z})$ and, assuming that the inverse matrix of $\mathbf{C}$ exists, then

$$\mathbf{A}\mathbf{C} = \mathbf{I} \Longrightarrow \mathbf{A} = \mathbf{C}^{-1} \Longrightarrow a_l^k c_i^l = \delta_{ki} \tag{4.18}$$

Introducing these $a_l^k(\mathbf{z})$ multipliers in equation (4.11) we obtain

$$u_k(\mathbf{z}) + \int_\Gamma u_i(\mathbf{x})\, \underset{\raise2pt\hbox{$\circ$}}{\underline{Q}}{}_i^k(\mathbf{x};\mathbf{z})\, \mathrm{d}\Gamma - \int_\Gamma q_i(\mathbf{x})\, \underset{\raise2pt\hbox{$\circ$}}{\underline{U}}{}_i^k(\mathbf{x};\mathbf{z})\, \mathrm{d}\Gamma =$$
$$\sum_j^R \alpha_j^p \left[\hat{u}_{jk}^p(\mathbf{z}) + \int_\Gamma \hat{u}_{ji}^p(\mathbf{x})\, \underset{\raise2pt\hbox{$\circ$}}{\underline{Q}}{}_i^k(\mathbf{x};\mathbf{z})\, \mathrm{d}\Gamma - \int_\Gamma \hat{q}_{ji}^p(\mathbf{x})\, \underset{\raise2pt\hbox{$\circ$}}{\underline{U}}{}_i^k(\mathbf{x};\mathbf{z})\, \mathrm{d}\Gamma \right] \tag{4.19}$$

where

$$\underset{\raise2pt\hbox{$\circ$}}{\underline{Q}}{}_i^k(\mathbf{x};\mathbf{z}) = a_l^k(\mathbf{z})\, \overset{0}{Q}{}_i^l(\mathbf{x};\mathbf{z}) \qquad \underset{\raise2pt\hbox{$\circ$}}{\underline{U}}{}_i^k(\mathbf{x};\mathbf{z}) = a_l^k(\mathbf{z})\, \overset{0}{U}{}_i^m(\mathbf{x};\mathbf{z}) \tag{4.20}$$

For the construction of the equation related to $u_{k,l}(\mathbf{z})$ a different approach is used and equation (4.11), where the integrals must be understood in the sense of Cauchy principal value, is not taken as starting point.

Starting from the general integral equation particularized for a point $z \in \Omega$, and taking the reference of the previous chapter, it can be written

$$u_m(\mathbf{z}) + \int_\Gamma u_i(\mathbf{x}) \overset{0}{Q}_i^m(\mathbf{x};\mathbf{z}) \, \mathrm{d}\Gamma = \int_\Gamma q_i(\mathbf{x}) \overset{0}{U}_i^m(\mathbf{x};\mathbf{z}) \, \mathrm{d}\Gamma + \int_\Omega \hat{b}_i(\mathbf{x}) \overset{0}{U}_i^m(\mathbf{x};\mathbf{z}) \, \mathrm{d}\Omega$$

$$(4.21)$$

where, in this case, no operation or simplification has been performed in the integrals. Given this, the derivatives of the above equation with respect to n are

$$u_{m,n}(\mathbf{z}) + \int_\Gamma u_i(\mathbf{x}) \overset{0}{Q}_{i,n}^m(\mathbf{x};\mathbf{z}) \, \mathrm{d}\Gamma = \int_\Gamma q_i(\mathbf{x}) \overset{0}{U}_{i,n}^m(\mathbf{x};\mathbf{z}) \, \mathrm{d}\Gamma +$$
$$\int_\Omega \hat{b}_i(\mathbf{x}) \overset{0}{U}_{i,n}^m(\mathbf{x};\mathbf{z}) \, \mathrm{d}\Omega$$

$$(4.22)$$

If the limit to the boundary of this equation is performed, and following the procedure shown in the previous chapter, we can get to an expression of type

$$d_{st}^{mn}(\mathbf{z}) u_{s,t}(\mathbf{z}) + \int_\Gamma u_i(\mathbf{x}) \overset{0}{Q}_{i,n}^m(\mathbf{x};\mathbf{z}) \, \mathrm{d}\Gamma = \int_\Gamma q_i(\mathbf{x}) \overset{0}{U}_{i,n}^m(\mathbf{x};\mathbf{z}) \, \mathrm{d}\Gamma +$$
$$\int_\Omega \hat{b}_i(\mathbf{x}) \overset{0}{U}_{i,n}^m(\mathbf{x};\mathbf{z}) \, \mathrm{d}\Omega$$

$$(4.23)$$

where the integrals must be understood in the sense of Hadamard finite parts and the free terms are grouped into $d_{mn}^{st}(\mathbf{z})$ coefficients. This equation has not been deduced generically for any fundamental solution but, in subsequent sections (see chapter 5), the deduction of the the particular expression using the analog operator selected in this work is shown.

The expansion of the independent term by means of approximation functions, combined with the transfer to the boundary of the volume integral using dual reciprocity techniques gives

$$d_{st}^{mn}\left(\mathbf{z}\right)u_{s,t}\left(\mathbf{z}\right) + \int_{\Gamma} u_i\left(\mathbf{x}\right)\overset{0}{Q}_{i,n}^{m}\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma - \int_{\Gamma} q_i\left(\mathbf{x}\right)\overset{0}{U}_{i,n}^{m}\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma$$

$$\sum_{j}^{R}\alpha_j^{p}\left[d_{st}^{mn}\left(\mathbf{z}\right)\hat{u}_{js,t}^{p}\left(\mathbf{z}\right) + \int_{\Gamma}\hat{u}_{ji}^{p}\left(\mathbf{x}\right)\overset{0}{Q}_{i,n}^{m}\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma - \int_{\Gamma}\hat{q}_{ji}^{p}\left(\mathbf{x}\right)\overset{0}{U}_{i,n}^{m}\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma\right]$$

$$(4.24)$$

In a similar way as the previous section, the coefficients $e_{mn}^{kl}\left(\mathbf{z}\right)$ are calculated for each point $\mathbf{z}$ using

$$e_{mn}^{kl}\left(\mathbf{z}\right)d_{st}^{mn}\left(\mathbf{z}\right) = \delta_{sk}\delta_{tl} \qquad (4.25)$$

The introduction of these multipliers $e_{mn}^{kl}\left(\mathbf{z}\right)$ in equation (4.24) produces

$$u_{k,l}\left(\mathbf{z}\right) + \int_{\Gamma} u_i\left(\mathbf{x}\right)\underline{\underline{\overset{0}{Q}}}_{i,l}^{k}\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma - \int_{\Gamma} q_i\left(\mathbf{x}\right)\underline{\underline{\overset{0}{U}}}_{i,l}^{k}\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma$$

$$(4.26)$$

$$\sum_{j}^{R}\alpha_j^{p}\left[\hat{u}_{jk,l}^{p}\left(\mathbf{z}\right) + \int_{\Gamma}\hat{u}_{ji}^{p}\left(\mathbf{x}\right)\underline{\underline{\overset{0}{Q}}}_{i,l}^{k}\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma - \int_{\Gamma}\hat{q}_{ji}^{p}\left(\mathbf{x}\right)\underline{\underline{\overset{0}{U}}}_{i,l}^{k}\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma\right]$$

where, respectively

$$\underline{\underline{\overset{0}{Q}}}_{i,l}^{k}\left(\mathbf{x};\mathbf{z}\right) = e_{mn}^{kl}\left(\mathbf{z}\right)\overset{0}{Q}_{i,n}^{m}\left(\mathbf{x};\mathbf{z}\right) \qquad \underline{\underline{\overset{0}{U}}}_{i,l}^{k}\left(\mathbf{z}\right) = e_{mn}^{kl}\left(\mathbf{z}\right)\overset{0}{U}_{i,n}^{m}\left(\mathbf{x};\mathbf{z}\right) \qquad (4.27)$$

Combining equations (4.17), (4.19) and (4.26) the integral version of the boundary conditions is obtained

$$-\gamma_m\left(\mathbf{z}\right) + \int_\Gamma u_i\left(\mathbf{x}\right)\overset{0}{P}_i^m\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma - \int_\Gamma q_i\left(\mathbf{x}\right)\overset{0}{R}_i^m\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma =$$

$$\sum_j^R \alpha_j^p\left[\hat{\zeta}_{jm}^p\left(\mathbf{z}\right) + \int_\Gamma \hat{u}_{ji}^p\left(\mathbf{x}\right)\overset{0}{P}_i^m\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma - \int_\Gamma \hat{q}_{ji}^p\left(\mathbf{x}\right)\overset{0}{R}_i^m\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma\right]$$

(4.28)

where

$$-\gamma_m\left(\mathbf{z}\right) = f_k^m\left(\mathbf{z}\right)u_k\left(\mathbf{z}\right) + g_{kl}^m\left(\mathbf{z}\right)u_{k,l}\left(\mathbf{z}\right)$$

$$\overset{0}{P}_i^m\left(\mathbf{x};\mathbf{z}\right) = f_k^m\left(\mathbf{z}\right)\underline{\overset{0}{Q}}_i^k\left(\mathbf{x};\mathbf{z}\right) + g_{kl}^m\left(\mathbf{z}\right)\underline{\underline{\overset{0}{Q}}}_{i,l}^k\left(\mathbf{x};\mathbf{z}\right)$$

$$\overset{0}{R}_i^m\left(\mathbf{x};\mathbf{z}\right) = f_k^m\left(\mathbf{z}\right)\underline{\overset{0}{U}}_i^k\left(\mathbf{x};\mathbf{z}\right) + g_{kl}^m\left(\mathbf{z}\right)\underline{\underline{\overset{0}{U}}}_{i,l}^k\left(\mathbf{x};\mathbf{z}\right)$$

$$\hat{\zeta}_{jm}^p\left(\mathbf{z}\right) = f_k^m\left(\mathbf{z}\right)\hat{u}_{jk}^p\left(\mathbf{z}\right) + g_{kl}^m\left(\mathbf{z}\right)\hat{u}_{jk,l}^p\left(\mathbf{z}\right)$$

(4.29)

4.5 General discretized system of equations

In subsequent chapters (see chapter 6) the detailed procedure of discretization and assembly of the resulting system using boundary elements will be discussed. In this chapter, we focus on the general case, a more abstract and compact[5] procedure, as described in the previous chapter[6], is used.

The aim of this section is to show that the set of integral equations defined in the previous sections allows, after the discretization procedure, to construct a determined linear system of equations able to solve the problem defined by (4.3a) and (4.3b).

Let consider, again, a generic problem involving a vector field $\mathbf{u}$ of range d. First, a functional approximation (by an approximating functions series) of the field $\mathbf{u}$ is defined. This approximation relates the value of $\mathbf{u}$ in the

---

[5]Noticed that, formally, this approach includes discretization schemes using elements.
[6]Mathematical conditions required for the approximation functions are not discussed to keep things simple in the analysis.

boundary with the value of a series of $\mathbf{M}$ approximation nodes located on $\mathbf{x}_l$.

$$u_i\left(\mathbf{x}\right) \approx \sum_{l}^{\mathbf{M}} \psi_l^i\left(\mathbf{x}\right) u_i(\mathbf{x}_l) \quad \text{for} \quad i = 1, d^7 \qquad (4.30)$$

In practice, these functions $\psi_l^i\left(\mathbf{x}\right)$ have non-zero values only in the vicinity of $\mathbf{x}_l$.

Similarly we proceed with $\mathbf{q}$ obtaining

$$q_i\left(\mathbf{x}\right) \approx \sum_{l}^{\mathbf{M}} \psi_l^i\left(\mathbf{x}\right) q_i(\mathbf{x}_l) \quad \text{for} \quad i = 1, d^7 \qquad (4.31)$$

Without loss of generality and taking into account the expression of boundary conditions, it can be assumed that the unknowns are

- $\mathbf{M} \times d$ unknowns associated with $u_i(\mathbf{x}_l)$

- $\mathbf{M} \times d$ unknowns associated with $q_i(\mathbf{x}_l)$

- $\mathbf{R} \times d$ unknowns associated with α_j^p

for a total of $(2\mathbf{M} + \mathbf{R}) \times d$ unknowns.

The next step is to introduce the expressions (4.30) and (4.31) in the integral equations. Analyzing one integral of equation (4.11) and performing the indicated transformation we obtain

$$\int_\Gamma u_i\left(\mathbf{x}\right) \overset{0}{Q}_i^m\left(\mathbf{x}; \mathbf{z}\right) d\Gamma \approx \sum_{l}^{\mathbf{M}} u_i(\mathbf{x}_l) \int_\Gamma \psi_l^i\left(\mathbf{x}\right) \overset{0}{Q}_i^m\left(\mathbf{x}; \mathbf{z}\right) d\Gamma \qquad (4.32)$$

---

[7]This equation does not follow the Einstein convention so the subscript i does not imply summation.

Thus, in the case of a collocation node k, with index l, located on $\mathbf{z}_k$, associated with an approximation function $\psi_l^i(\mathbf{x})$ and an approximation node located on $\mathbf{x}_l$ we define $\bar{h}_{ij}$ as

$$\bar{h}_{ij}(l; \mathbf{z}_k) = \int_\Gamma \psi_l^i(\mathbf{x}) \overset{0}{Q}_i^j(\mathbf{x}; \mathbf{z}_k) \, d\Gamma \quad \text{for} \quad i, j = 1, d \qquad (4.33)$$

The above expression does not follow the Einstein convention so the index i does not imply summation.

Defining $\overline{\mathbf{H}}_{kl}$ as the matrix of range $d \times d$ whose components are $\bar{h}_{ij}(l; \mathbf{z}_k)$ and $\mathbf{u}_l$ as the vector whose components are $u_i(\mathbf{x}_l)$, equation (4.32) can be written as

$$\int_\Gamma u_i(\mathbf{x}) \overset{0}{Q}_i^m(\mathbf{x}; \mathbf{z}_k) \, d\Gamma \approx \sum_l^{\mathbf{M}} \overline{\mathbf{H}}_{kl} \mathbf{u}_l \qquad (4.34)$$

Following a similar procedure is concluded that

$$\int_\Gamma q_i(\mathbf{x}) \overset{0}{U}_i^m(\mathbf{x}; \mathbf{z}_k) \, d\Gamma \approx \sum_l^{\mathbf{M}} \mathbf{G}_{kl} \mathbf{q}_l \qquad (4.35)$$

The term $c_i^m(\mathbf{z}_k) u_i(\mathbf{z}_k)$ is analyzed similarly. If equation (4.30) is considered we can write

$$c_i^m(\mathbf{z}_k) u_i(\mathbf{z}_k) \approx \sum_l c_i^m(\mathbf{z}_k) \psi_l^i(\mathbf{z}_k) u_i(\mathbf{x}_l) \qquad (4.36)$$

Focusing on a node located on $\mathbf{z}_k$, the matrix $\mathbf{C}_{kl}$ of dimension $d \times d$ can be constructed. Each component of this matrix is defined as

$$c_{ij}(l; \mathbf{z}_k) = c_j^i(\mathbf{z}_k) \psi_l^j(\mathbf{z}_k) \quad \text{para} \quad i, j = 1, d \qquad (4.37)$$

thus

$$c_i^m \left(\mathbf{z}_k \right) u_i \left(\mathbf{z}_k \right) \approx \sum_{l}^{\mathbf{M}} \mathbf{C}_{kl} \mathbf{u}_l \qquad \boxed{4.38}$$

Taking into account the previous results, the left-hand side of equation (4.11), particularized on a collocation node located on $\mathbf{z}_k$ is

$$c_i^m \left(\mathbf{z}_k \right) u_i \left(\mathbf{z}_k \right) + \int_{\Gamma} u_i \left(\mathbf{x} \right) \overset{0}{Q}_i^m \left(\mathbf{x}; \mathbf{z}_k \right) \mathrm{d}\Gamma - \int_{\Gamma} q_i \left(\mathbf{x} \right) \overset{0}{U}_i^m \left(\mathbf{x}; \mathbf{z}_k \right) \mathrm{d}\Gamma \qquad \boxed{4.39}$$

is transformed into

$$\sum_{l}^{\mathbf{M}} \mathbf{C}_{kl} \mathbf{u}_l + \sum_{l}^{\mathbf{M}} \overline{\mathbf{H}}_{kl} \mathbf{u}_l - \sum_{l}^{\mathbf{M}} \mathbf{G}_{kl} \mathbf{q}_l = \sum_{l}^{\mathbf{M}} \mathbf{H}_{kl} \mathbf{u}_l - \sum_{l}^{\mathbf{M}} \mathbf{G}_{kl} \mathbf{q}_l \qquad \boxed{4.40}$$

Several simplifications, that reduce the computation time very significantly[8], can be used in the calculation of the second part of the integral in equation (4.11).

If (4.11) expression is analyzed

$$\sum_{j}^{\mathbf{R}} \alpha_j^p \left[c_i^m \left(\mathbf{z} \right) \hat{u}_{ji}^p \left(\mathbf{z} \right) + \int_{\Gamma} \hat{u}_{ji}^p \left(\mathbf{x} \right) \overset{0}{Q}_i^m \left(\mathbf{x}; \mathbf{z} \right) \mathrm{d}\Gamma - \int_{\Gamma} \hat{q}_{ji}^p \left(\mathbf{x} \right) \overset{0}{U}_i^m \left(\mathbf{x}; \mathbf{z} \right) \mathrm{d}\Gamma \right] \qquad \boxed{4.41}$$

Unlike in the previous case, all functions appearing in the integrals are known and the only unknowns are the α_j^p coefficients. In this case, therefore, the integrals can be calculated "exactly". However, if the required numerical effort for the "exact" calculation is compared to the simplification indicated in (4.40), the effort is $\mathbf{R} \times d$ times higher using the same integration parameters.

---

[8]In the previous chapter, these simplifications were introduced directly, for sake of brevity and because of its common use in the literature.

To avoid this, we can approximate $\hat{\mathbf{u}}$ and $\hat{\mathbf{q}}$ with the same functional decomposition used for the fields $\mathbf{u}$ and $\mathbf{q}$. Taking this into account and "freezing" index j, we get

$$\alpha_j^p \left[c_i^m \left(\mathbf{z}_k \right) \hat{u}_{ji}^p \left(\mathbf{z}_k \right) + \int_\Gamma \hat{u}_{ji}^p \left(\mathbf{x} \right) \overset{0}{Q}_i^m \left(\mathbf{x}; \mathbf{z}_k \right) \mathrm{d}\Gamma - \int_\Gamma \hat{q}_{ji}^p \left(\mathbf{x} \right) \overset{0}{U}_i^m \left(\mathbf{x}; \mathbf{z}_k \right) \mathrm{d}\Gamma \right] \quad \boxed{4.42}$$

can be calculated using the same procedure as the first member of equation (4.11), getting

$$\boldsymbol{\alpha}_j \left[\sum_l^{\mathbf{M}} \mathbf{H}_{kl} \hat{\mathbf{u}}_{jl} - \sum_l^{\mathbf{M}} \mathbf{G}_{kl} \hat{\mathbf{q}}_{jl} \right] \quad \boxed{4.43}$$

Summing with respect to index j gives

$$\sum_j^{\mathbf{R}} \boldsymbol{\alpha}_j \left[\sum_l^{\mathbf{M}} \mathbf{H}_{kl} \hat{\mathbf{u}}_{jl} - \sum_l^{\mathbf{M}} \mathbf{G}_{kl} \hat{\mathbf{q}}_{jl} \right] = \left[\sum_l^{\mathbf{M}} \mathbf{H}_{kl} \hat{\mathbf{U}}_l - \sum_l^{\mathbf{M}} \mathbf{G}_{kl} \hat{\mathbf{Q}}_l \right] \boldsymbol{\alpha} \quad \boxed{4.44}$$

where, respectively, $\hat{\mathbf{U}}_l$ and $\hat{\mathbf{Q}}_l$ are matrices with vectors $\hat{\mathbf{u}}_{jl}$ and $\hat{\mathbf{q}}_{jl}$ as columns. $\boldsymbol{\alpha}$ is column vector containing all the α_j^p unknowns.

Taking the above into account and using (4.39), equation (4.11) for a collocation node located on $\mathbf{z}_k \in \Gamma$ can be written as

$$\sum_l^{\mathbf{M}} \mathbf{H}_{kl} \mathbf{u}_l - \sum_l^{\mathbf{M}} \mathbf{G}_{kl} \mathbf{q}_l = \left[\sum_l^{\mathbf{M}} \mathbf{H}_{kl} \hat{\mathbf{U}}_l - \sum_l^{\mathbf{M}} \mathbf{G}_{kl} \hat{\mathbf{Q}}_l \right] \boldsymbol{\alpha} \quad \boxed{4.45}$$

The sizes of the blocks of this equation are

- $\mathbf{u}_l$, $\mathbf{q}_l$ dimension $d \times 1$

- $\mathbf{H}_{kl}$, $\mathbf{G}_{kl}$ dimension $d \times d$

- $\hat{\mathbf{U}}_l, \hat{\mathbf{Q}}_l$ dimension $d \times R \cdot d$

- $\boldsymbol{\alpha}$ dimension $R \cdot d \times 1$

If this equation is applied to a collection of **M** collocation nodes k located on the boundary and the index l associated with **M** approximation nodes are contracted, the discretized version of the **u-BIE** matrix of equations is obtained

$$\boxed{\mathbf{Hu} - \mathbf{Gq} - \left[\mathbf{H\hat{U}} - \mathbf{G\hat{Q}}\right]\boldsymbol{\alpha} = 0} \qquad (4.46)$$

In this case the sizes of the blocks of this equation are

- $\mathbf{u}$, $\mathbf{q}$ dimension $M \cdot d \times 1$

- $\mathbf{H}$, $\mathbf{G}$ dimension $M \cdot d \times M \cdot d$

- $\hat{\mathbf{U}}, \hat{\mathbf{Q}}$ dimension $M \cdot d \times R \cdot d$

- $\boldsymbol{\alpha}$ dimension $R \cdot d \times 1$

- Globally we have a system of $M \cdot d$ equations

To obtain the matrix version of equation **b-BIE** a similar procedure is followed. For a node located in $\mathbf{z}_k \in \Omega$ it can be written

$$-\mathbf{b}_k + \sum_l^{\mathbf{M}} \mathbf{K}_{kl}\mathbf{u}_l - \sum_l^{\mathbf{M}} \mathbf{L}_{kl}\mathbf{q}_l = \left[\hat{\mathbf{B}}_k + \sum_l^{\mathbf{M}} \mathbf{K}_{kl}\hat{\mathbf{U}}_l - \sum_l^{\mathbf{M}} \mathbf{L}_{kl}\hat{\mathbf{Q}}_l\right]\boldsymbol{\alpha} \qquad (4.47)$$

where $\hat{\mathbf{B}}_k$ is a matrix with vectors $\hat{\boldsymbol{\beta}}_{jk}$ as columns and $\mathbf{b}_k$ is column vector containing the values of the independent term α_j^p, both particularized in $\mathbf{z}_k \in \Omega$.

If this equation is applied to a collection of **R** internal nodes k located in the domain and the index l associated with **M** approximation nodes is contracted, the discretized version of the **b-BIE** matrix of equations is obtained

$$\boxed{\mathbf{Ku} - \mathbf{Lq} - \left[\hat{\mathbf{B}} + \mathbf{K}\hat{\mathbf{U}} - \mathbf{L}\hat{\mathbf{Q}}\right]\boldsymbol{\alpha} = \mathbf{b}} \qquad \text{(4.48)}$$

In this case the sizes of the blocks of this equation are

- $\mathbf{u}$, $\mathbf{q}$ dimension $M \cdot d \times 1$
- $\mathbf{K}$, $\mathbf{L}$ dimension $R \cdot d \times M \cdot d$
- $\hat{\mathbf{U}}$, $\hat{\mathbf{Q}}$, $\hat{\mathbf{B}}$ dimension $R \cdot d \times R \cdot d$
- $\boldsymbol{\alpha}$ dimension $R \cdot d \times 1$
- $\mathbf{b}$ dimension $R \cdot d \times 1$
- Globally we have a system of $R \cdot d$ equations

Finally, to obtain the matrix version of equation **q-BIE** the same procedure is performed again. Thus, for a node located in $\mathbf{z}_k \in \Gamma$ it can be written

$$-\boldsymbol{\gamma}_k + \sum_l^{\mathbf{M}} \mathbf{P}_{kl}\mathbf{u}_l - \sum_l^{\mathbf{M}} \mathbf{R}_{kl}\mathbf{q}_l = \left[\hat{\mathbf{E}}_k + \sum_l^{\mathbf{M}} \mathbf{P}_{kl}\hat{\mathbf{U}}_l - \sum_l^{\mathbf{M}} \mathbf{L}_{kl}\hat{\mathbf{R}}_l\right]\boldsymbol{\alpha} \qquad \text{(4.49)}$$

where $\hat{\mathbf{E}}_k$ is a matrix with vectors $\hat{\boldsymbol{\zeta}}_{jk}$ as columns and $\boldsymbol{\gamma}_k$ is column vector containing the values of the independent term $\boldsymbol{\gamma}(\mathbf{z})$, both particularized on $\mathbf{z}_k \in \Gamma$.

If this equation is applied to a collection of **M** collocation nodes k located on the boundary and the index l associated with **M** approximation nodes is contracted, the discretized version of the **q-BIE** matrix of equations is obtained

$$\boxed{\mathbf{Pu} - \mathbf{Rq} - \left[\hat{\mathbf{E}} + \mathbf{P}\hat{\mathbf{U}} - \mathbf{R}\hat{\mathbf{Q}}\right]\boldsymbol{\alpha} = \boldsymbol{\gamma}} \qquad \text{(4.50)}$$

In this case the sizes of the blocks of this equation are

- **u**, **q** dimension $M \cdot d \times 1$

- **P**, **R** dimension $M \cdot d \times M \cdot d$

- $\hat{\mathbf{U}}$, $\hat{\mathbf{Q}}$, $\hat{\mathbf{E}}$ dimension $M \cdot d \times R \cdot d$

- $\boldsymbol{\alpha}$ dimension $R \cdot d \times 1$

- $\boldsymbol{\gamma}$ dimension $M \cdot d \times 1$

- Globally we have a system of $M \cdot d$ equations

The system formed by equations (4.46), (4.48), (4.50) has $(2\mathbf{M} + \mathbf{R}) \times d$ equations and $(2\mathbf{M} + \mathbf{R}) \times d$ unknowns. This determined linear system of equations allows to solve the problem posed initially. This system can be written in matrix form as

$$\mathbf{AX} = \mathbf{B} \qquad (4.51)$$

where, without loss of generality, the vector $\mathbf{X}$ contains the unknowns associated with the values of $\mathbf{u}$ and $\mathbf{q}$ at the approximation nodes located on $\mathbf{z}_k$, and the values of the $\boldsymbol{\alpha}$ coefficients which describe the unknown independent term that appears due to the application of the analog operator.

Truth acquired by thinking of our own is like a natural limb; it alone really belongs to us.

Arthur Schopenhauer

Particularized Analog Equation

In the previous chapter, a general algorithm based on Boundary Element Method and Analog Equation Method has been developed. This algorithm has been analyzed generically, without further assessment on the selected analog operator, the problem to solve or the type of approximation functions.

In this chapter the peculiarities of problem to be solved and the simplifications associated with the analog operator chosen are presented.

5.1 Laplace Operator as Analog Operator

Although several works, based on different types of analog operators (e.g. Wang uses the homogeneous isotropic elastic operator in [116]), can be found in the literature, Laplace operator is the most usual choice. It was used in both the original works (see [60]) and most subsequent developments (see e.g. [61]).

In this work this choice is maintained, due to the simplicity of the associated fundamental solution, which will prevent further complications in the mathematical developments and, especially, because it produces uncoupled expressions, simplifying considerably the design and reducing computational load.

Taking the analog equation of the problem as starting point, we have

$$\nabla^2 u_i + \hat{b}_i\left(\mathbf{x}\right) = 0 \quad \text{in} \quad \mathbf{x} \in \Omega \quad i = 1, d \qquad \boxed{5.1\text{a}}$$

$$\mathcal{G}_i(\mathbf{u}) + \gamma_i\left(\mathbf{x}\right) = 0 \quad \text{on} \quad \mathbf{x} \in \Gamma \quad i = 1, d \qquad \boxed{5.1\text{b}}$$

Notice that, at this point, no limitation regarding the dimension of the problem have been introduced. The problem can be either two or three dimensional, thermoelastic, poroeleastic... because the formulation is formally equal. It is assumed that $\overset{0}{U}$ is the fundamental solution associated with a scalar Laplace problem defined in a domain with the same dimension as in our original problem. Since we are dealing with a vectorial problem of dimension d, the following equation must be solved in order to obtain the fundamental solution

$$\nabla^2 u_i + \delta_{im}\delta\left(\mathbf{x} - \mathbf{z}\right) = 0 \quad \text{for} \quad i = 1, d \qquad \boxed{5.2}$$

Clearly, if $i = m$ the problem is reduced to the calculation of the fundamental solution of the scalar Laplace operator, therefore $u_i = \overset{0}{U}$. On the other hand if $i \neq m$, then $u_i = 0$. Thus it is found that the problem is uncoupled and, the loads applied in the m direction only generate responses in the same direction. Taking this into account

$$\overset{0}{U}{}^m_i = \overset{0}{U}\delta_{mi} \qquad \boxed{5.3}$$

The introduction of this fundamental solution in the basic general integral equation (prior to perform the limit to the boundary)

$$u_m\left(\mathbf{z}\right) + \int_\Gamma u_i\left(\mathbf{x}\right)\delta_{mi}\overset{0}{Q}\left(\mathbf{x}; \mathbf{z}\right)\mathrm{d}\Gamma = \int_\Gamma q_i\left(\mathbf{x}\right)\delta_{mi}\overset{0}{U}\left(\mathbf{x}; \mathbf{z}\right)\mathrm{d}\Gamma +$$
$$\int_\Omega \hat{b}_i\left(\mathbf{x}\right)\delta_{mi}\overset{0}{U}\left(\mathbf{x}; \mathbf{z}\right)\mathrm{d}\Omega \qquad \boxed{5.4}$$

produces

$$u_m\left(\mathbf{z}\right) + \int_\Gamma u_m\left(\mathbf{x}\right) \overset{0}{Q}\left(\mathbf{x}; \mathbf{z}\right) d\Gamma = \int_\Gamma q_m\left(\mathbf{x}\right) \overset{0}{U}\left(\mathbf{x}; \mathbf{z}\right) d\Gamma + \int_\Omega \hat{b}_m\left(\mathbf{x}\right) \overset{0}{U}\left(\mathbf{x}; \mathbf{z}\right) d\Omega$$

$$(5.5)$$

This equation is identical to the equation of the scalar problem, so the particularization on a boundary point is

$$c\left(\mathbf{z}\right) u_m\left(\mathbf{z}\right) + \int_\Gamma u_m\left(\mathbf{x}\right) \overset{0}{Q}\left(\mathbf{x}; \mathbf{z}\right) d\Gamma = \int_\Gamma q_m\left(\mathbf{x}\right) \overset{0}{U}\left(\mathbf{x}; \mathbf{z}\right) d\Gamma + \int_\Omega \hat{b}_m\left(\mathbf{x}\right) \overset{0}{U}\left(\mathbf{x}; \mathbf{z}\right) d\Omega$$

$$(5.6)$$

The simplification of the terms associated with the expansion of the unknown independent term $\hat{b}_m\left(\mathbf{x}\right)$, can be performed immediately following the same procedure.

Introducing the functional approximation of $\hat{b}_m\left(\mathbf{x}\right)$ by R approximation functions

$$\hat{b}_m\left(\mathbf{x}\right) \approx \sum_j^R \alpha_j^m f_j\left(\mathbf{x}\right)$$

$$(5.7)$$

where, in this case, the functions $f_j\left(\mathbf{x}\right)$ are chosen so that analytical solutions of the analog operator, with this set of functions as source terms, exist

$$\nabla^2 u_m + f_j\left(\mathbf{x}\right) \mathbf{e}_l = 0 \quad \text{for} \quad m = 1, d$$

$$(5.8)$$

Since the equations are uncoupled, it is evident that the solution to these equations vanishes except when $m = l$. In this case

$$\hat{\mathbf{u}}_j^m\left(\mathbf{x}\right) = \hat{u}_j\left(\mathbf{x}\right) \mathbf{e}_m \Longrightarrow \hat{u}_{ji}^m = \hat{u}_j \delta_{mi}$$
$$\hat{\mathbf{q}}_j^m\left(\mathbf{x}\right) = \hat{q}_j\left(\mathbf{x}\right) \mathbf{e}_m \Longrightarrow \hat{q}_{ji}^m = \hat{q}_j \delta_{mi}$$

$$(5.9)$$

Introducing these expressions in equation (4.11), the integral equation **u-BIE**, simplified using Laplace operator as analog operator, is obtained.

$$
\begin{aligned}
c\left(\mathbf{z}\right) u_m\left(\mathbf{z}\right) + \int_\Gamma u_m\left(\mathbf{x}\right) \overset{0}{Q}\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma - \int_\Gamma q_m\left(\mathbf{x}\right) \overset{0}{U}\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma = \\
\sum_j^R \alpha_j^m \left[c\left(\mathbf{z}\right) \hat{u}_j\left(\mathbf{z}\right) + \int_\Gamma \hat{u}_j\left(\mathbf{x}\right) \overset{0}{Q}\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma - \int_\Gamma \hat{q}_j\left(\mathbf{x}\right) \overset{0}{U}\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma \right]
\end{aligned}
$$

(5.10)

where this equation is extended to indices $m = 1, d$.

For the construction of **b-BIE** equation, the equation (4.15), which is repeated for clarity of exposition, is used as starting point

$$
\begin{aligned}
-b_m\left(\mathbf{z}\right) + \int_\Gamma u_i\left(\mathbf{x}\right) \overset{0}{K}_i^m\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma - \int_\Gamma q_i\left(\mathbf{x}\right) \overset{0}{L}_i^m\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma = \\
\sum_j^R \alpha_j^p \left[\hat{\beta}_{jm}^p\left(\mathbf{z}\right) + \int_\Gamma \hat{u}_{ji}^p\left(\mathbf{x}\right) \overset{0}{K}_i^m\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma - \int_\Gamma \hat{q}_{ji}^p\left(\mathbf{x}\right) \overset{0}{L}_i^m\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma \right]
\end{aligned}
$$

(5.11)

where a change of notation from the original equation has been introduced.

$$
\begin{aligned}
-b_m = \mathcal{L}_m(\mathbf{u}) && \hat{\beta}_{jm}^p = \mathcal{L}_m(\hat{\mathbf{u}}_j^p) \\
\overset{0}{K}_i^m = \mathcal{L}_m(\overset{0}{\mathbf{Q}}_i) && \overset{0}{L}_i^m = \mathcal{L}_m(\overset{0}{\mathbf{U}}_i)
\end{aligned}
$$

(5.12)

To obtain the above expressions, it is necessary to proceed as follows. In this simplified case the vectors $\overset{0}{\mathbf{Q}}_i$ and $\overset{0}{\mathbf{U}}_i$ are, respectively

$$
\overset{0}{\mathbf{U}}_i = \overset{0}{U}\delta_{mi}\mathbf{e}_m = \overset{0}{U}\mathbf{e}_i \qquad \overset{0}{\mathbf{Q}}_i = \overset{0}{Q}\delta_{mi}\mathbf{e}_m = \overset{0}{Q}\mathbf{e}_i
$$

(5.13)

Since the differential operator $\mathcal{L}$ is linear, we can define, without loss of generality, the operator $\mathcal{L}_m^i$ as the part of the m component of the operator $\mathcal{L}$, which affects the component i of the vector to which $\mathcal{L}$ is applied. Introducing this notation we obtain

$$\overset{o}{K}{}^{m}_{i} = \mathcal{L}_m(\overset{o}{\mathbf{Q}}_i) = \mathcal{L}^i_m(\overset{o}{Q}) \qquad \overset{o}{L}{}^{m}_{i} = \mathcal{L}_m(\overset{o}{\mathbf{U}}_i) = \mathcal{L}^i_m(\overset{o}{U}) \tag{5.14}$$

The term $\hat{\beta}^p_{jm}$ can be deduced in a similar way. Taking into account that, in this case

$$\hat{\mathbf{u}}^p_j(\mathbf{x}) = \hat{u}_j(\mathbf{x})\,\mathbf{e}_p \Longrightarrow \hat{\beta}^p_{jm} = \mathcal{L}_m(\hat{\mathbf{u}}^p_j) = \mathcal{L}^p_m(\hat{u}_j) \tag{5.15}$$

If the simplifications indicated in (5.14), (5.15) and (5.9) are introduced in equation (5.11), **b-BIE** particularized equation is obtained.

$$
\begin{aligned}
&-b_m(\mathbf{z}) + \int_\Gamma u_i(\mathbf{x})\,\mathcal{L}^i_m\left[\overset{o}{Q}(\mathbf{x};\mathbf{z})\right]\mathrm{d}\Gamma - \int_\Gamma q_i(\mathbf{x})\,\mathcal{L}^i_m\left[\overset{o}{U}(\mathbf{x};\mathbf{z})\right]\mathrm{d}\Gamma = \\
&\sum_j^R \alpha^i_j\left\{\mathcal{L}^i_m\left[\hat{u}_j(\mathbf{z})\right] + \int_\Gamma \hat{u}_j(\mathbf{x})\,\mathcal{L}^i_m\left[\overset{o}{Q}(\mathbf{x};\mathbf{z})\right]\mathrm{d}\Gamma - \int_\Gamma \hat{q}_j(\mathbf{x})\,\mathcal{L}^i_m\left[\overset{o}{U}(\mathbf{x};\mathbf{z})\right]\mathrm{d}\Gamma\right\}
\end{aligned}
\tag{5.16}
$$

In order to obtain the particularized **q-BIE** equation, the same procedure, as in the previous chapter is followed. The generic boundary condition was defined as

$$\mathcal{G}_m(\mathbf{u}) = h^m_k(\mathbf{z})\,u_k + g^m_{kl}(\mathbf{z})\,u_{k.l} \qquad \text{where} \quad \mathbf{z} \in \Gamma \tag{5.17}$$

The boundary equation associated with $u_k(\mathbf{z})$ is obtained directly from (5.10)

$$
\begin{aligned}
&u_k(\mathbf{z}) + \int_\Gamma u_k(\mathbf{x})\,\underline{\overset{o}{Q}}(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma - \int_\Gamma q_k(\mathbf{x})\,\underline{\overset{o}{U}}(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma = \\
&\sum_j^R \alpha^k_j\left[\hat{u}_j(\mathbf{z}) + \int_\Gamma \hat{u}_j(\mathbf{x})\,\underline{\overset{o}{Q}}(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma - \int_\Gamma \hat{q}_j(\mathbf{x})\,\underline{\overset{o}{U}}(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma\right]
\end{aligned}
\tag{5.18}
$$

where in this case

$$\overset{0}{\underline{Q}}(\mathbf{x};\mathbf{z}) = \frac{\overset{0}{Q}(\mathbf{x};\mathbf{z})}{c(\mathbf{z})} \qquad \overset{0}{\underline{U}}(\mathbf{x};\mathbf{z}) = \frac{\overset{0}{U}(\mathbf{x};\mathbf{z})}{c(\mathbf{z})} \tag{5.19}$$

In order to obtain the term $u_{k,l}(\mathbf{z})$, the basic BEM equation, without limit to the boundary or expansion of the independent term, is used as starting point. So we have

$$u_k(\mathbf{z}) + \int_\Gamma u_k(\mathbf{x}) \overset{0}{Q}(\mathbf{x};\mathbf{z})\, \mathrm{d}\Gamma = \int_\Gamma q_k(\mathbf{x}) \overset{0}{U}(\mathbf{x};\mathbf{z})\, \mathrm{d}\Gamma +$$
$$\int_\Omega \hat{b}_k(\mathbf{x}) \overset{0}{U}(\mathbf{x};\mathbf{z})\, \mathrm{d}\Omega \tag{5.20}$$

A generic derivative of the above equation, with respect to x_n coordinate, is obtained. Notice again that this equation, although is formally vectorial, is uncoupled so it can be reduced to d identical equations. With this in mind, the analysis of the restriction of the derivative on a boundary point located on $\mathbf{z}$, can be done by studying the integral equation associated with the scalar problem.

Appendices (A.2) of this work contain the deduction of this limit for two and three dimensional problems. At this point of the analysis, a generic formulation is kept for simplicity. The integrals must be understood in the sense of Cauchy principal value (or Hadamard finite part) and the free terms are grouped in $d_t^n(\mathbf{z})$ coefficients. The values of these coefficients are deduced in the appendix and can be extracted from equations (A.40) and (A.70).

For three-dimensional problems the coefficients are

$$d_t^n(\mathbf{z}) = \frac{1}{2}\delta_{nt} - \int_\Gamma \frac{r_{,n}}{4\pi r^2}\left[n_t(\mathbf{z}) - n_t(\mathbf{x})\right] - \oint_\Gamma \frac{\mathbf{e}_n \wedge \mathbf{e}_t}{4\pi r}\mathrm{d}\mathbf{l} \tag{5.21}$$

And, respectively, in two-dimensional problems

$$d_t^n(\mathbf{z}) = \frac{1}{2}\delta_{nt} - \int_\Gamma \frac{r_{,n}}{2\pi r}\left[n_t(\mathbf{z}) - n_t(\mathbf{x})\right] - \epsilon_{nt}\int_\Gamma \frac{\boldsymbol{\nabla} r}{2\pi}\,\mathrm{d}\mathbf{l} \qquad (5.22)$$

In compact form, the derivative of the equation (5.20) can be written as

$$d_t^n(\mathbf{z})\,u_{k,t}(\mathbf{z}) + \int_\Gamma u_k(\mathbf{x})\,\overset{0}{Q}_{,n}(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma = \int_\Gamma q_k(\mathbf{x})\,\overset{0}{U}_{,n}(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma +$$
$$\int_\Omega \hat{b}_k(\mathbf{x})\,\overset{0}{U}_{,n}(\mathbf{x};\mathbf{z})\,\mathrm{d}\Omega \qquad (5.23)$$

where, again a set coefficients $e_n^l(\mathbf{z})$ must be deduced. These coefficients satisfy

$$e_n^l(\mathbf{z})\,d_t^n(\mathbf{z}) = \delta_{tl} \qquad (5.24)$$

Introducing these multipliers $e_n^l(\mathbf{z})$ in the previous equation, and expanding the independent term similarly to the previous sections, results in

$$u_{k,l}(\mathbf{z}) + \int_\Gamma u_k(\mathbf{x})\,\underline{\underline{\overset{0}{Q}}}_l(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma - \int_\Gamma q_k(\mathbf{x})\,\underline{\underline{\overset{0}{U}}}_l(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma =$$
$$\sum_j^R \alpha_j^k\left[\hat{u}_{j,l}(\mathbf{z}) + \int_\Gamma \hat{u}_j(\mathbf{x})\,\underline{\underline{\overset{0}{Q}}}_l(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma - \int_\Gamma \hat{q}_j(\mathbf{x})\,\underline{\underline{\overset{0}{U}}}_l(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma\right] \qquad (5.25)$$

where

$$\underline{\underline{\overset{0}{Q}}}_l(\mathbf{x};\mathbf{z}) = e_n^l(\mathbf{z})\,\overset{0}{Q}_{,n}(\mathbf{x};\mathbf{z}) \qquad \underline{\underline{\overset{0}{U}}}_l(\mathbf{x};\mathbf{z}) = e_n^l(\mathbf{z})\,\overset{0}{U}_{,n}(\mathbf{x};\mathbf{z}) \qquad (5.26)$$

The combination of (5.18) and (5.25) with equation (5.17) leads to

$$\boxed{\begin{aligned} -\gamma_m(\mathbf{z}) + \int_\Gamma u_i(\mathbf{x})\,\overset{0}{P}_i^m(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma - \int_\Gamma q_i(\mathbf{x})\,\overset{0}{R}_i^m(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma = \\ \sum_j^R \alpha_j^i\left[\hat{\zeta}_{jm}^i(\mathbf{z}) + \int_\Gamma \hat{u}_j(\mathbf{x})\,\overset{0}{P}_i^m(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma - \int_\Gamma \hat{q}_j(\mathbf{x})\,\overset{0}{R}_i^m(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma\right] \end{aligned}} \qquad (5.27)$$

where

$$-\gamma_m\left(\mathbf{z}\right) = h_k^m\left(\mathbf{z}\right) u_k\left(\mathbf{z}\right) + g_{kl}^m\left(\mathbf{z}\right) u_{k,l}\left(\mathbf{z}\right)$$

$$\overset{0}{P_i^m}\left(\mathbf{x};\mathbf{z}\right) = h_i^m\left(\mathbf{z}\right)\underline{\overset{0}{Q}}\left(\mathbf{x};\mathbf{z}\right) + g_{il}^m\left(\mathbf{z}\right)\underline{\underline{\overset{0}{Q}}}_l\left(\mathbf{x};\mathbf{z}\right)$$

$$\overset{0}{R_i^m}\left(\mathbf{x};\mathbf{z}\right) = h_i^m\left(\mathbf{z}\right)\underline{\overset{0}{U}}\left(\mathbf{x};\mathbf{z}\right) + g_{il}^m\left(\mathbf{z}\right)\underline{\overset{0}{U}}_l\left(\mathbf{x};\mathbf{z}\right)$$

$$\hat{\zeta}_{jm}^i\left(\mathbf{z}\right) = h_i^m\left(\mathbf{z}\right)\hat{u}_j\left(\mathbf{z}\right) + g_{il}^m\left(\mathbf{z}\right)\hat{u}_{j,l}\left(\mathbf{z}\right)$$

$$\boxed{5.28}$$

Equations (5.10), (5.16) and (5.28) form a set of integral equations, which after the discretization procedure allows to solve the problem formulated. This set of discretized equations is identical to that calculated in the previous chapter, with some simplifications due to the use of the fundamental solution associated with the Laplace operator. Before proceeding to detail the process of discretization of these equations, the particularization to the elastic problem is analyzed. This is because one purpose of this thesis is the use of this numerical methodology for solving elastic problems.

5.2 Elastostatics Problem

In this section the set of integral equations obtained previously is particularized for the elastic problem. In particular, in what follows the algorithm is restricted to the static linear elastic case. Likewise, it is assumed that the boundary conditions are given in the form of displacements or tensions (and not as combinations).

Two particularized formulations are developed. The first includes the more general definition of properties, covering inhomogeneous anisotropic materials while the latter is restricted to the case of inhomogeneous isotropic materials.

The formulation of the problem defined by equations (4.3a) and (4.3b) can be easily obtained in a large number of textbooks and is easily deduced from

Equilibrium equations

$$\sigma_{ij,j} + b_i = 0 \qquad \boxed{5.29}$$

Constitutive equations

$$\sigma_{ij} = C_{ijkl}\left(\mathbf{x}\right)\varepsilon_{kl} \qquad \boxed{5.30}$$

The coefficients C_{ijkl} define the elasticity tensor. For general anisotropic materials, these coefficients are independent functions only related by constraints equations $C_{ijkl} = C_{ijlk} = C_{jikl} = C_{klij}$ [1]. In the case of isotropic materials the coefficients are defined as

$$C_{ijkl} = \lambda\left(\mathbf{x}\right)\delta_{ij}\delta_{kl} + \mu\left(\mathbf{x}\right)\left(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}\right) \qquad \boxed{5.31}$$

and, consequently, the constitutive equations are

$$\sigma_{ij} = \lambda\left(\mathbf{x}\right)\delta_{ij}\varepsilon_{kk} + 2\mu\left(\mathbf{x}\right)\varepsilon_{ij} \qquad \boxed{5.32}$$

Strain-displacement equations

$$\varepsilon_{kl} = \frac{1}{2}\left(u_{k,l} + u_{l.k}\right) \qquad \boxed{5.33}$$

Traction

$$t_i = \sigma_{ij}n_j \qquad \boxed{5.34}$$

Combining these expressions[2] we obtain

---

[1]These constraints are due to symmetries of the stress and strain tensors and the invariance of the energy of deformation.

[2]Where it has taken into account that $C_{ijkl} = C_{ijlk}$.

$$C_{ijkl}\left(\mathbf{x}\right)u_{k,lj} + C_{ijkl,j}\left(\mathbf{x}\right)u_{k,l} + b_i\left(\mathbf{x}\right) = 0 \quad \text{in } \mathbf{x} \in \Omega \quad i = 1, d \qquad \text{(5.35a)}$$

$$u_i = \overline{u}_i\left(\mathbf{x}\right) \quad \text{on } \mathbf{x} \in \Gamma_u \qquad \text{(5.35b)}$$

$$t_i = C_{ijkl}\left(\mathbf{x}\right)u_{k,l}n_j = \overline{t}_i\left(\mathbf{x}\right) \quad \text{on } \mathbf{x} \in \Gamma_t \qquad \text{(5.35c)}$$

where equation (5.35a), in the case of an isotropic material is

$$\begin{aligned} &(\lambda\left(\mathbf{x}\right) + \mu\left(\mathbf{x}\right))u_{j,ji} + \mu\left(\mathbf{x}\right)u_{i,jj} + \lambda_{,i}\left(\mathbf{x}\right)u_{j,j} + \\ &\mu_{,j}\left(\mathbf{x}\right)(u_{i,j} + u_{j,i}) + b_i\left(\mathbf{x}\right) = 0 \quad \text{en } \mathbf{x} \in \Omega \quad i = 1, d \end{aligned} \qquad \text{(5.36)}$$

and, respectively, equation (5.35c) is

$$t_i = \lambda\left(\mathbf{x}\right)u_{k,k}n_i + \mu\left(\mathbf{x}\right)(u_{i,j} + u_{j,i})n_j = \overline{t}_i\left(\mathbf{x}\right) \quad \text{en } \mathbf{x} \in \Gamma_t \qquad \text{(5.37)}$$

The introduction of these expressions in the integral equation results in

u-BIE equation

In **u-BIE** equation there is no change associated with the particularization, so the equation is identical to (5.10). It is repeated for clarity of exposition.

$$\begin{aligned} &c\left(\mathbf{z}\right)u_m\left(\mathbf{z}\right) + \int_\Gamma u_m\left(\mathbf{x}\right)\overset{0}{Q}\left(\mathbf{x}; \mathbf{z}\right)\mathrm{d}\Gamma - \int_\Gamma q_m\left(\mathbf{x}\right)\overset{0}{U}\left(\mathbf{x}; \mathbf{z}\right)\mathrm{d}\Gamma = \\ &\sum_j^R \alpha_j^m \left[c\left(\mathbf{z}\right)\hat{u}_j\left(\mathbf{z}\right) + \int_\Gamma \hat{u}_j\left(\mathbf{x}\right)\overset{0}{Q}\left(\mathbf{x}; \mathbf{z}\right)\mathrm{d}\Gamma - \int_\Gamma \hat{q}_j\left(\mathbf{x}\right)\overset{0}{U}\left(\mathbf{x}; \mathbf{z}\right)\mathrm{d}\Gamma\right] \end{aligned} \qquad \text{(5.38)}$$

b-BIE equation

Formally, **b-BIE** equation remains unaltered. There are only changes focused on the definitions of the kernels. Thus, equation (5.16) is repeated

$$-b_m\left(\mathbf{z}\right) + \int_\Gamma u_i\left(\mathbf{x}\right)\mathcal{L}_m^i\left[\overset{0}{Q}\left(\mathbf{x};\mathbf{z}\right)\right]\mathrm{d}\Gamma - \int_\Gamma q_i\left(\mathbf{x}\right)\mathcal{L}_m^i\left[\overset{0}{U}\left(\mathbf{x};\mathbf{z}\right)\right]\mathrm{d}\Gamma =$$

$$\sum_j^R \alpha_j^i\left\{\mathcal{L}_m^i\left[\hat{u}_j\left(\mathbf{z}\right)\right] + \int_\Gamma \hat{u}_j\left(\mathbf{x}\right)\mathcal{L}_m^i\left[\overset{0}{Q}\left(\mathbf{x};\mathbf{z}\right)\right]\mathrm{d}\Gamma - \int_\Gamma \hat{q}_j\left(\mathbf{x}\right)\mathcal{L}_m^i\left[\overset{0}{U}\left(\mathbf{x};\mathbf{z}\right)\right]\mathrm{d}\Gamma\right\}$$

$$(5.39)$$

In this case, it is possible to give explicit expressions of $\mathcal{L}_m^i$ because every involved operators is analytically defined. If equation (5.35a), which uses the classical subscripts, is compared with its integral version defined by (5.39), an adjustment is necessary in these subscripts for coupling both expressions and avoid duplication of indices. So the following index changes will be introduced in (5.35a): $i \to m$, $k \to i$ and $j \to k$.

Taking this into account, the explicit expression of, for example, $\hat{u}_j$ is

$$\mathcal{L}_m^i\left[\hat{u}_j\right] = C_{mkil}\left[\hat{u}_{j,lk}\right] + C_{mkil,k}\left[\hat{u}_{j,l}\right] \qquad (5.40)$$

Similar expressions for $\overset{0}{U}$ and $\overset{0}{Q}$ can be obtained following the same procedure.

In the case of an isotropic material, we can operate similarly, and, simply the introduction of (5.31) in the above expressions results in

$$\mathcal{L}_m^i\left[\hat{u}_j\right] = (\lambda + \mu)\hat{u}_{j,mi} + \mu\delta_{mi}\hat{u}_{j,kk} + \lambda_{,m}\hat{u}j, i + \mu_{,k}\delta_{mi}\hat{u}_{j,k} + \mu_{,i}\hat{u}_{j,m} \qquad (5.41)$$

and again equivalent expressions can be deduced for $\overset{0}{U}$ and $\overset{0}{Q}$.

q-BIE equation

As in the previous section, **q-BIE** equation does not present, at formal level, any difference with the general equation, except direct particularization of the independent term as tractions. Thereby adapting (5.28)

$$
\begin{aligned}
t_m\left(\mathbf{z}\right) + \int_\Gamma u_i\left(\mathbf{x}\right) \overset{0}{P}_i^m\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma - \int_\Gamma q_i\left(\mathbf{x}\right) \overset{0}{R}_i^m\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma = \\
\sum_j^R \alpha_j^i \left[\hat{\zeta}_{jm}^i\left(\mathbf{z}\right) + \int_\Gamma \hat{u}_j\left(\mathbf{x}\right) \overset{0}{P}_i^m\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma - \int_\Gamma \hat{q}_j\left(\mathbf{x}\right) \overset{0}{R}_i^m\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma \right]
\end{aligned}
$$

$$(5.42)$$

If the boundary conditions notation of the general case, defined by (5.17) is used and, considering that the boundary conditions that equation **q-BIE** represents are tractions, it is immediately concluded that, in the case of a general material

$$
f_k^m\left(\mathbf{z}\right) = 0 \qquad g_{kl}^m\left(\mathbf{z}\right) = C_{mpkl}\left(\mathbf{z}\right) n_p
$$

$$(5.43)$$

and in the homogeneous and isotropic

$$
f_k^m\left(\mathbf{z}\right) = 0 \qquad g_{kl}^m\left(\mathbf{z}\right) = \lambda\left(\mathbf{z}\right) \delta_{kl} n_m + \mu\left(\mathbf{z}\right) \left(\delta_{mk} n_l + \delta_{ml} n_k\right)
$$

$$(5.44)$$

Given the above, the system of equations defined in (5.28) can be particularized for elastostatic problems obtaining

$$
\begin{aligned}
\overset{0}{P}_i^m\left(\mathbf{x};\mathbf{z}\right) &= C_{mpil}\left(\mathbf{z}\right) n_p\left(\mathbf{z}\right) \overset{0}{\underline{\underline{Q}}}_l\left(\mathbf{x};\mathbf{z}\right) \\
\overset{0}{R}_i^m\left(\mathbf{x};\mathbf{z}\right) &= C_{mpil}\left(\mathbf{z}\right) n_p\left(\mathbf{z}\right) \overset{0}{\underline{\underline{U}}}_l\left(\mathbf{x};\mathbf{z}\right) \\
\hat{\zeta}_{jm}^i\left(\mathbf{z}\right) &= C_{mpil}\left(\mathbf{z}\right) n_p\left(\mathbf{z}\right) \hat{u}_{j,l}\left(\mathbf{z}\right)
\end{aligned}
$$

$$(5.45)$$

and, respectively

$$\overset{0}{P_i^m}(\mathbf{x};\mathbf{z}) = \lambda(\mathbf{z})\underline{\underline{\overset{0}{Q}}}_i(\mathbf{x};\mathbf{z})\,n_m(\mathbf{z}) + \mu(\mathbf{z})\left(\delta_{mi}\underline{\underline{\overset{0}{Q}}}_l(\mathbf{x};\mathbf{z})\,n_l(\mathbf{z}) + \underline{\underline{\overset{0}{Q}}}_m(\mathbf{x};\mathbf{z})\,n_i(\mathbf{z})\right)$$

$$\overset{0}{R_i^m}(\mathbf{x};\mathbf{z}) = \lambda(\mathbf{z})\underline{\overset{0}{U}}_i(\mathbf{x};\mathbf{z})\,n_m(\mathbf{z}) + \mu(\mathbf{z})\left(\delta_{mi}\underline{\overset{0}{U}}_l(\mathbf{x};\mathbf{z})\,n_l(\mathbf{z}) + \underline{\overset{0}{U}}_m(\mathbf{x};\mathbf{z})\,n_i(\mathbf{z})\right)$$

$$\hat{\zeta}_{jm}^i(\mathbf{z}) = \lambda(\mathbf{z})\hat{u}_{j,i}(\mathbf{z})\,n_m(\mathbf{z}) + \mu(\mathbf{z})\left(\delta_{mi}\hat{u}_{j,l}(\mathbf{z})\,n_l(\mathbf{z}) + \hat{u}_{j,m}(\mathbf{z})\,n_i(\mathbf{z})\right)$$

$$(5.46)$$

From this point, the procedure is identical to the general one for both cases and it is merely repeated for clarity of presentation. First of all, the coefficients $d_t^n(\mathbf{z})$, which only depend on the chosen analog operator[3], are calculated. The value of these coefficients in two and three-dimensional problems is given respectively by (5.21) and (5.22).

After obtaining the coefficients $d_t^n(\mathbf{z})$, the coefficients $e_n^l(\mathbf{z})$ can be deduced using the following equation

$$e_n^l(\mathbf{z})\,d_t^n(\mathbf{z}) = \delta_{tl} \tag{5.47}$$

And again, these coefficients allow to relate the terms $\overset{0}{Q}_l$ and $\underline{\overset{0}{U}}_l$, listed in (5.46), with the fundamental solution by the equation

$$\underline{\underline{\overset{0}{Q}}}_l(\mathbf{x};\mathbf{z}) = e_n^l(\mathbf{z})\,\overset{0}{Q}_{,n}(\mathbf{x};\mathbf{z}) \qquad \underline{\overset{0}{U}}_l(\mathbf{x};\mathbf{z}) = e_n^l(\mathbf{z})\,\overset{0}{U}_{,n}(\mathbf{x};\mathbf{z}) \tag{5.48}$$

The set of integral equations defined by (5.38), (5.42) and (5.39) allows, after the discretization process, the build a linear system of equations able to solve the elastostatic problem.

The generic discretization procedure would be similar to that presented in the previous chapter, and its formulation is not included for the sake of brevity. Instead, in the next chapter of this thesis, a more detailed analysis

---

[3]The procedure to obtain the coefficients is detailed in the appendices.

of the chosen discretization procedure, based on boundary elements, will be performed.

6

Discretization and assembly of the system of equations

In the previous chapter, the system of particularized integral equations has been obtained. Two simplifications were introduced to particularize the equations. First, Laplace operator is used as analog operator and second, the problem selected to be solved is an elastic problem, for both anisotropic and isotropic case, involving homogeneous and inhomogeneous materials.

From this point, the analysis is focused exclusively on the case of isotropic materials (homogeneous or inhomogeneous). Relevant aspects of the procedure of implementation and discretization of integral equations are detailed. This procedure leads to obtain the algebraic system of equations that solves the problem.

6.1 Discretization using Boundary Elements

The discretization by means of boundary elements is the classical technique, that names the Boundary Element Method, used in order to construct the algebraic system of equations that allows to solve numerically the boundary value problems formulated in sets of Boundary Integral Equations. This

technique, whose details can be found in numerous texts as [34], is included in the generic technique introduced in (3.1.4).

This procedure is discussed without going into great detail. Integral equation **u-BIE** of the scalar potential problem is taking as starting point. Assuming that the collocation points are on a "smooth" part of the boundary **u-BIE** equation is written as

$$\frac{1}{2}u\left(\mathbf{z}\right) + \int_{\Gamma} u\left(\mathbf{x}\right) \overset{\mathrm{o}}{Q}\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma = \int_{\Gamma} q\left(\mathbf{x}\right) \overset{\mathrm{o}}{U}\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma \qquad \boxed{6.1}$$

The domain of integration Γ is divided into a series of NE elements Γ_k so that The discretization procedure consists in the division of the boundary into a series of elements. In these elements the variables of the problem to be studied (in this case u y q) are approximated by the discrete value (sometimes unknown) at the so-called approximation nodes[1], and a set of shape functions whose domain of definition is restricted to a single element.

Thus, dividing the boundary into N elements, equation (6.1) is transformed into

$$\frac{1}{2}u\left(\mathbf{z}\right) + \sum_{j=1}^{N} \int_{\Gamma_j} u\left(\mathbf{x}\right) \overset{\mathrm{o}}{Q}\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma_j = \sum_{j=1}^{N} \int_{\Gamma_j} q\left(\mathbf{x}\right) \overset{\mathrm{o}}{U}\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma_j \qquad \boxed{6.2}$$

On the other hand, for an element j with P approximation nodes we can write

$$u \approx \sum_{l=1}^{P} \phi_l u_l^j \qquad\qquad q \approx \sum_{l=1}^{P} \phi_l q_l^j \qquad \boxed{6.3}$$

where it has been considered that the approximation functions, formally, do not vary among the different elements because its definition is given in

---

[1]Commonly, these approximation nodes can belong to more than one element.

natural coordinates. This natural coordinates are associated with different types of elements.

The introduction of (6.3) in the integrals of equation (6.2) results, for the first term, in

$$\int_{\Gamma_j} u(\mathbf{x}) \overset{0}{Q}(\mathbf{x}; \mathbf{z}) \, d\Gamma_j = \sum_{l=1}^{P} u_l^j \int_{\Gamma_j} \phi_l(\mathbf{x}) \overset{0}{Q}(\mathbf{x}; \mathbf{z}) \, d\Gamma_j \qquad (6.4)$$

where, from this point, equalities are used instead of approximations. If the above equation is particularized for a collocation node located on coordinates $\mathbf{z}_i$ we can define

$$\hat{h}_{il}^j = \int_{\Gamma_j} \phi_l(\mathbf{x}) \overset{0}{Q}(\mathbf{x}; \mathbf{z}_i) \, d\Gamma_j \qquad g_{il}^j = \int_{\Gamma_j} \phi_l(\mathbf{x}) \overset{0}{U}(\mathbf{x}; \mathbf{z}_i) \, d\Gamma_j \qquad (6.5)$$

and consequently

$$\int_{\Gamma_j} u(\mathbf{x}) \overset{0}{Q}(\mathbf{x}; \mathbf{z}_i) \, d\Gamma_j = \sum_{l=1}^{P} \hat{h}_{il}^j u_l^j \qquad (6.6)$$

and

$$\int_{\Gamma_j} q(\mathbf{x}) \overset{0}{U}(\mathbf{x}; \mathbf{z}_i) \, d\Gamma_j = \sum_{l=1}^{P} g_{il}^j q_l^j \qquad (6.7)$$

In order to evaluate the integrals of the terms $\hat{h}_{il}^j$ y g_{il}^j, it should be noted that the analog equation selected in this work is the Laplace equation. This means that the integration algorithms, used to evaluate these integrals, possess a relatively simple formulation (even in singularities) and are available in the literature. The details of these algorithms are omitted in this work and can be found in several textbooks such as [34].

CHAPTER 6. DISCRETIZATION AND ASSEMBLY OF THE SYSTEM OF EQUATIONS

A few aspects, for purposes of the numerical implementation of these algorithms, must be mentioned

- An isoparametric representation is used, so the same function is being used to approximate both the geometry of the elements and the variables of the problem.

- The evaluation of regular integrals is performed using standard Gauss quadratures.

- The evaluation of the near-singular integrals are performed by standard Gauss quadrature, but using an adaptive algorithm with a tree scheme. This means that the domain of integration is divided into subdomains and the integral is calculated in the domain and as a sum of the integrals of the subdomains. Comparing these values, the "quality" of the evaluation of the integral can be studied and, eventually, the recursive application of this algorithm achieves the prescribed error level.

- The singularities of the fundamental solution of Laplace, are only weakly singular and its calculation does not involve great difficulty. Several algorithms are available in the literature, i.e. [34].

- Two types of second order elements are implemented in this work: six node triangles and nine node quadrilaterals[2]

Introducing the coefficients $\hat{h}_{il}^j$ and g_{il}^j in equation (6.2) and particularizing it on a point located on $\mathbf{z}_i$, we get

$$\frac{1}{2}u\left(\mathbf{z}_i\right) + \sum_{j=1}^{N}\sum_{l=1}^{P}\hat{h}_{il}^j u_l^j = \sum_{j=1}^{N}\sum_{l=1}^{P}g_{il}^j q_l^j \qquad \boxed{6.8}$$

where the first term of the equation $(\frac{1}{2}u\left(\mathbf{z}_i\right))$ has remained unchanged for

---

[2]Formally, the formulation is similar regardless of the type of element, although there are some peculiarities that introduce slight differences in the integration algorithms.

now, and has not been approximated based on nodal values.

The summations defined on equation (6.8), cover all the elements and, for each element, the approximation nodes that are included in it. In order to compact this equation, the above two summations can be mixed in a single sum that goes through all the approximation nodes of the boundary Γ. Assuming there are NA approximation nodes, equation (6.8) can be written as

$$\frac{1}{2} u\left(\mathbf{z}_i\right) + \sum_{k=1}^{NA} \hat{h}_{ik} u_k = \sum_{k=1}^{NA} g_{ik} q_k \qquad (6.9)$$

where new coefficients $\hat{h}_{ik}$ and g_{ik} have been introduced. The index k refers to the approximation node.

Assuming the approximation node with index k is part of M elements, $\hat{h}_{ik}$ is defined as

$$\hat{h}_{ik} = \sum_{j}^{M} \int_{\Gamma_j} \phi_l\left(\mathbf{x}\right) \overset{o}{Q}\left(\mathbf{x}; \mathbf{z}_i\right) \mathrm{d}\Gamma_j = \sum_{j}^{M} \hat{h}_{il}^{j} \qquad (6.10)$$

This sum includes the contributions from the M elements where the approximation node is included. In this sense, ϕ_l should be understood as the shape function that is applied to the approximation node k, when we are integrating over an element j that includes k.

The last step to compact the formulation implies to deal with the first term. Assuming that the collocation node located on $\mathbf{z}_i$ is included in an element j, the application of (6.3) results in[3]

$$u\left(\mathbf{z}_i\right) = \sum_{l=1}^{P} \phi_l\left(\mathbf{z}_i\right) u_l^{j} \qquad (6.11)$$

---

[3]The coordinates of the node $\mathbf{z}_i$ are introduced into the function ϕ_l via its natural coordinates in the element.

CHAPTER 6. DISCRETIZATION AND ASSEMBLY OF THE SYSTEM OF EQUATIONS

Keeping in mind that, due to the type of approximation functions used in this work, every variable defined on a point located on $\mathbf{z}_i$, always can be described unambiguously as a unique function of the approximation nodes, no matter if it belongs to one element or to several elements. This uniqueness is easy to check if we realize that a point located on $\mathbf{z}_i$ can

1. Be contained within an element or an edge belonging to a single element. In this case it is clear that any variable is calculated in terms of the nodal values of that element.

2. Be contained within the boundary of an element belonging to several elements. In this case, due to the definition of the shape functions used, the resulting expression is identical regardless of the element, since it only depends on the nodes of the common edge. The definition of the shape functions used and their properties can be found in [34]. For example, if $\mathbf{z}_i$ matches a approximation node p then, $\phi_l = 1$ if $l = p$ and $\phi_l = 0$ if $l \neq p$.

Taking this into account, we can define a discrete function $a(i, k)$ such that, for a pair constituted by a collocation node i located on $\mathbf{z}_i$ and a approximation node k

$$a(i,k) = a_{ik} = \begin{cases} 0 & \text{if } k \text{ does not belong to an element that contains } i \\ \phi_l & \text{if } k \text{ belongs to an element that contains } i \end{cases}$$

$\boxed{6.12}$

where, as indicated, function ϕ_l is unique for each approximation node and its value depends on the natural coordinates of the collocation node i in the element.

Introducing (6.12) in (6.9) we get

$$\frac{1}{2}\sum_{k=1}^{NA} a_{ik} u_k\left(\mathbf{z}_i\right) + \sum_{k=1}^{NA} \hat{h}_{ik} u_k = \sum_{k=1}^{NA} g_{ik} q_k$$

$\boxed{6.13}$

Defining h_{ik} as

$$h_{ik} = \hat{h}_{ik} + \frac{1}{2}a_{ik}$$

(6.14)

thus it can finally be written as

$$\sum_{k=1}^{NA} h_{ik}u_k = \sum_{k=1}^{NA} g_{ik}q_k$$

(6.15)

If this equation is applied to a collection of M collocation nodes located on the boundary, the discretized version of the system of equations is obtained. This system can be written, in matrix form, as follows

$$\mathbf{Hu} = \mathbf{Gq}$$

(6.16)

6.2 Discretization of the algorithm AEM-BEM

The implementation of the integral equations, at programming level, as were presented in the previous chapter, has two drawbacks from a practical point of view. Firstly, if standard methodology for the construction of the system of equations is followed, the explicit expression of the kernels (associated with the fundamental solution of the Laplace operator and its derivatives) are used to evaluate the integrals.

And secondly, if the **q-BIE** equation is analyzed, it is easy to check that the kernels expressions $\overset{0}{P}_i^m(\mathbf{x};\mathbf{z})$ and $\overset{0}{R}_i^m(\mathbf{x};\mathbf{z})$ of equation (5.46) depends, through the equations (5.48) and (5.47), on the coefficients $d_t^n(\mathbf{z})$ (defined in (5.21)) and the properties of the material at the collocation node. These coefficients are calculable but not known "a priori". They are calculated on runtime, since they depend on the geometry of the problem.

Similarly, the kernels of the **b-BIE** equations depend, through equation (5.41), on the properties of materials in the internal collocation nodes where the equation is applied.

The algorithm is intended to be used in three-dimensional problems and, consequently, there is an exponential increase in computational load. This condition discards[4] the use of languages with advanced features that allow, naturally, using functions created at runtime (e.g, Matlab) and limits the range of efficient implementation to FORTRAN and C. In this work, object-oriented FORTRAN [51] is the language selected to implement the AEM-BEM algorithm. It keeps intact the benefits, in terms of efficient code generation, introducing object-oriented capabilities[5].

The implementation, in a static type language, of "a priori" unknown functions requires the use of artifices that significantly complicate the implementation process.

On the other hand, the use of the AEM methodology allows flexibility in the application of the algorithm to different types of problem, and an object-oriented language allows to generate a more reusable and modular software easily scalable to deal with new problems with different differential operators. Considering these two aspects, the implementation of a particular kernel valid only for a particular operator (such as elastic) partially penalizes the advantages of the method.

To avoid this problem, it is sufficient to notice that, by using the Laplace operator as analog operator, the equations are implicitly uncoupled. Therefore, the integral equations of any linear problem can be decomposed into linear combinations of integral equations (which are independent of the type of problem) multiplied by coefficients that depend only on **z**.

Considering these aspects, we will proceed to detail the computational

---

[4]Aspects of simplicity of parallelization or problems associated with memory bandwidth are not analyzed. This must be understood in the sense of generating efficient monocore code.

[5]This topic will be discussed in more detail in the next chapter of this thesis.

implementation of the AEM-BEM method.

As described in previous chapters, the system is divided in three main blocks associated with different equations.

6.2.1 Block u-BIE

If equation (5.38) is analyzed, it is evident that the equation is uncoupled and is identical for each index m. The deduction of the **u-BIE** equations block is reduced to the calculation of d scalar problems, where d is the dimension of the problem to solve.

So, ignoring the subscript m, and using R approximation functions for the term $\hat{b}$ we can write

$$
\begin{aligned}
&c\left(\mathbf{z}\right) u\left(\mathbf{z}\right) + \int_{\Gamma} u\left(\mathbf{x}\right) \overset{\text{o}}{Q}\left(\mathbf{x}; \mathbf{z}\right) \mathrm{d}\Gamma - \int_{\Gamma} q\left(\mathbf{x}\right) \overset{\text{o}}{U}\left(\mathbf{x}; \mathbf{z}\right) \mathrm{d}\Gamma = \\
&\sum_{j}^{R} \alpha_j \left[c\left(\mathbf{z}\right) \hat{u}_j\left(\mathbf{z}\right) + \int_{\Gamma} \hat{u}_j\left(\mathbf{x}\right) \overset{\text{o}}{Q}\left(\mathbf{x}; \mathbf{z}\right) \mathrm{d}\Gamma - \int_{\Gamma} \hat{q}_j\left(\mathbf{x}\right) \overset{\text{o}}{U}\left(\mathbf{x}; \mathbf{z}\right) \mathrm{d}\Gamma \right]
\end{aligned}
$$

(6.17)

Whose discretized version for a collocation node i and M approximation nodes is

$$
\sum_{k}^{M} h_{ik} u_k - \sum_{k}^{M} g_{ik} q_k = \sum_{j}^{R} \alpha_j \left[\sum_{k}^{M} h_{ik} \hat{u}_{kj} - \sum_{k}^{M} g_{ik} \hat{q}_{kj} \right]
$$

(6.18)

It should noted again that, in terms of implementation, in this works is considered that the boundary is "smooth" around the collocation points, so that

$$
c\left(\mathbf{z}\right) = \frac{1}{2} \implies h_{ik} = \hat{h}_{ik} + \frac{1}{2} a_{ik}
$$

(6.19)

CHAPTER 6. DISCRETIZATION AND ASSEMBLY OF THE SYSTEM OF EQUATIONS

Also noticed that the integrals associated with the approximation functions of the independent term $\hat{b}_i$, for example

$$\int_\Gamma \hat{u}_j(\mathbf{x}) \overset{0}{Q}(\mathbf{x};\mathbf{z}) \, d\Gamma \qquad (6.20)$$

can be calculated exactly, because all functions involved are known explicitly.

However, in this work we have chosen to approximate the functions $\hat{u}_j$ and $\hat{q}_j$ using the same approximation scheme used in the variables u y q, because the computational load is greatly reduced. Briefly, let us assume that the computational cost associated with the integration of an element is defined as C. Taking into account there are NE elements, NC collocation nodes and NA approximation functions, the total cost of all integrals of the type depicted in equation (6.20), which are necessary to build the discretized system of equations, would be of order $C \times NE \times NC \times NA$. Using the shape functions, the terms h_{ik} y g_{ik} can be reused, and the computational cost is reduced to the evaluation of functions $\hat{u}_j$ y $\hat{q}_j$ on approximation nodes and a matrix product.

The vector equation is uncoupled so its construction can be immediately performed from the scalar equation. Thus, rearranging terms, it can be written to a collocation node i

$$\sum_k^M \mathbf{H}_{ik}\mathbf{u}_k - \sum_k^M \mathbf{G}_{ik}\mathbf{q}_k - \sum_j^R \boldsymbol{\alpha}_j \left[\sum_k^M \mathbf{H}_{ik}\hat{\mathbf{u}}_{kj} - \sum_k^M \mathbf{G}_{ik}\hat{\mathbf{q}}_{kj} \right] = 0 \qquad (6.21)$$

where matrices $\mathbf{H}_{ik}$ and $\mathbf{G}_{ik}$ are diagonal matrices of order d and every term of the identical has the same value.

$$\mathbf{H}_{ik} = h_{ik}\mathbf{I} \qquad \mathbf{G}_{ik} = g_{ik}\mathbf{I} \qquad (6.22)$$

And vectors $\mathbf{u}_k$, $\mathbf{q}_k$, $\boldsymbol{\alpha}_j$, $\hat{\mathbf{u}}_{kj}$ and $\hat{\mathbf{q}}_{kj}$ are of dimension d.

If this equation is applied to a collection of M collocation nodes located on the boundary, the discretized version of the **u-BIE** block of equations is obtained, which can be written in matrix form as

$$\boxed{\mathbf{Hu} - \mathbf{Gq} - \left[\mathbf{H\hat{U}} - \mathbf{G\hat{Q}}\right]\boldsymbol{\alpha} = 0} \qquad (6.23)$$

The sizes of the elements of this equation are

- $\mathbf{u}$, $\mathbf{q}$ dimension $M \cdot d \times 1$

- $\mathbf{H}$, $\mathbf{G}$ dimension $M \cdot d \times M \cdot d$

- $\mathbf{\hat{U}}, \mathbf{\hat{Q}}$ dimension $M \cdot d \times R \cdot d$

- $\boldsymbol{\alpha}$ dimension $R \cdot d \times 1$

6.2.2 Block b-BIE

For the calculation of the block **b-BIE**, the procedure described in paragraph (6.2) is taken into consideration. Thus, the integral equations of this block are decomposed into linear combinations of integral equations. This thesis is focused on the elastic operator, so the first and second order derivatives of the integral equation are required.

Again, it is taken into account that the Laplace operator is used as analog operator. This means that the integrals appear uncoupled and are identical for every index, allowing the use of the scalar equation to build the system of vector equations. Thus, using equation (6.17) for an internal point located in $\mathbf{z}_i$, which is repeated for clarity of presentation, as starting point

$$u(\mathbf{z}) + \int_{\Gamma} u(\mathbf{x})\overset{o}{Q}(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma - \int_{\Gamma} q(\mathbf{x})\overset{o}{U}(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma =$$
$$\sum_{j}^{R} \alpha_j \left[\hat{u}_j(\mathbf{z}) + \int_{\Gamma} \hat{u}_j(\mathbf{x})\overset{o}{Q}(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma - \int_{\Gamma} \hat{q}_j(\mathbf{x})\overset{o}{U}(\mathbf{x};\mathbf{z})\,\mathrm{d}\Gamma\right] \qquad (6.24)$$

the derivative of this equation with respect to $\dfrac{\partial}{\partial z_s}$ is performed obtaining

$$u_{,s}\left(\mathbf{z}\right) + \int_{\Gamma} u\left(\mathbf{x}\right) \overset{0}{Q}_{,s}\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma - \int_{\Gamma} q\left(\mathbf{x}\right) \overset{0}{U}_{,s}\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma =$$
$$\sum_{j}^{R} \alpha_j \left[\hat{u}_{j,s}\left(\mathbf{z}\right) + \int_{\Gamma} \hat{u}_j\left(\mathbf{x}\right) \overset{0}{Q}_{,s}\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma - \int_{\Gamma} \hat{q}_j\left(\mathbf{x}\right) \overset{0}{U}_{,s}\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma \right]$$

(6.25)

whose discretized version is

$$u_{i,s} + \sum_{k}^{M} \hat{h}_{ik,s} u_k - \sum_{k}^{M} g_{ik,s} q_k = \sum_{j}^{R} \alpha_j \left[\hat{u}_{ij,s} + \sum_{k}^{M} \hat{h}_{ik,s} \hat{u}_{kj} - \sum_{k}^{M} g_{ik,s} \hat{q}_{kj} \right]$$

(6.26)

Also the second derivative with respect to $\dfrac{\partial^2}{\partial z_s \partial z_t}$ is

$$u_{,st}\left(\mathbf{z}\right) + \int_{\Gamma} u\left(\mathbf{x}\right) \overset{0}{Q}_{,st}\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma - \int_{\Gamma} q\left(\mathbf{x}\right) \overset{0}{U}_{,st}\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma =$$
$$\sum_{j}^{R} \alpha_j \left[\hat{u}_{j,st}\left(\mathbf{z}\right) + \int_{\Gamma} \hat{u}_j\left(\mathbf{x}\right) \overset{0}{Q}_{,st}\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma - \int_{\Gamma} \hat{q}_j\left(\mathbf{x}\right) \overset{0}{U}_{,st}\left(\mathbf{x};\mathbf{z}\right) \mathrm{d}\Gamma \right]$$

(6.27)

whose discretized version is

$$u_{i,st} + \sum_{k}^{M} \hat{h}_{ik,st} u_k - \sum_{k}^{M} g_{ik,st} q_k = \sum_{j}^{R} \alpha_j \left[\hat{u}_{ij,st} + \sum_{k}^{M} \hat{h}_{ik,st} \hat{u}_{kj} - \sum_{k}^{M} g_{ik,st} \hat{q}_{kj} \right]$$

(6.28)

As it has been indicated, the vector problem is constructed from the equations of the scalar problem. For this, taking the case of a general material, the equation associated with the term b_i is

$$\sigma_{ij,j} + b_i = C_{ijkl,j} u_{k,l} + C_{ijkl} u_{k,lj} + b_i = 0$$

(6.29)

On the other hand, the discretized equation of type **b-BIE** associated with an internal point located in $\mathbf{z}_i$ is (see (4.47))

$$\sum_k^M \mathbf{K}_{ik}\mathbf{u}_k - \sum_k^M \mathbf{L}_{ik}\mathbf{q}_k - \sum_j^R \alpha_j \left[\hat{\mathbf{b}}_{ij} + \sum_k^M \mathbf{K}_{ik}\hat{\mathbf{u}}_{kj} - \sum_k^M \mathbf{L}_{ik}\hat{\mathbf{q}}_{kj} \right] = \mathbf{b}_i \quad (6.30)$$

Comparing equation (6.29) with equation (6.30), and taking into account the expressions of equations (6.26) and (6.28), the matrix $\mathbf{K}_{ik}$ is formulated as

$$\begin{bmatrix} C_{1j1l,j}\hat{h}_{ik,l} + C_{1j1l}\hat{h}_{ik,lj} & C_{1j2l,j}\hat{h}_{ik,l} + C_{1j2l}\hat{h}_{ik,lj} & C_{1j3l,j}\hat{h}_{ik,l} + C_{1j3l}\hat{h}_{ik,lj} \\ C_{2j1l,j}\hat{h}_{ik,l} + C_{2j1l}\hat{h}_{ik,lj} & C_{2j2l,j}\hat{h}_{ik,l} + C_{2j2l}\hat{h}_{ik,lj} & C_{2j3l,j}\hat{h}_{ik,l} + C_{2j3l}\hat{h}_{ik,lj} \\ C_{3j1l,j}\hat{h}_{ik,l} + C_{3j1l}\hat{h}_{ik,lj} & C_{3j2l,j}\hat{h}_{ik,l} + C_{3j2l}\hat{h}_{ik,lj} & C_{3j3l,j}\hat{h}_{ik,l} + C_{3j3l}\hat{h}_{ik,lj} \end{bmatrix}$$
$$(6.31)$$

So to calculate the matrix $\mathbf{K}_{ik}$, the following terms must be obtained

$$\begin{bmatrix} \hat{h}_{ik,1} \\ \hat{h}_{ik,2} \\ \hat{h}_{ik,3} \end{bmatrix} \quad \begin{bmatrix} \hat{h}_{ik,11} & \hat{h}_{ik,12} & \hat{h}_{ik,13} \\ \hat{h}_{ik,21} & \hat{h}_{ik,22} & \hat{h}_{ik,23} \\ \hat{h}_{ik,31} & \hat{h}_{ik,32} & \hat{h}_{ik,33} \end{bmatrix} \quad (6.32)$$

Remark again that these terms are independent of the differential operator governing the problem.

To obtain the terms $\mathbf{L}_{ik}$ and $\hat{b}_{ij}$ a similar procedure is followed.

Clearly, this approach has the advantage of maintaining the flexibility of the method dividing the calculation into two parts. The first one is the calculation of the integrals associated with the variables and their derivatives and is common to all problems and carries most of the computation time. The second one includes the multipliers associated with each particular operator which only depend on the coordinate $\mathbf{z}$.

If this equation is applied to a collection of R internal collocation nodes located inside the domain, the discretized version of the **b-BIE** block of equations is obtained, which can be written in matrix form as

$$\boxed{\mathbf{Ku} - \mathbf{Lq} - \left[\hat{\mathbf{B}} + \mathbf{K}\hat{\mathbf{U}} - \mathbf{L}\hat{\mathbf{Q}}\right]\boldsymbol{\alpha} = \mathbf{b}} \qquad (6.33)$$

The sizes of the elements of this equation are

- $\mathbf{u}$, $\mathbf{q}$ dimension $M \cdot d \times 1$

- $\mathbf{K}$, $\mathbf{L}$ dimension $R \cdot d \times M \cdot d$

- $\hat{\mathbf{U}}$, $\hat{\mathbf{Q}}$, $\hat{\mathbf{B}}$ dimension $R \cdot d \times R \cdot d$

- $\boldsymbol{\alpha}$ dimension $R \cdot d \times 1$

- $\mathbf{b}$ dimension $R \cdot d \times 1$

6.2.3   Block q-BIE

Finally, the block of discretized integral equations associated with the boundary conditions is constructed. The same aspects as in the previous section are considered, although certain peculiarities must be added.

For a general problem, the boundary integral equations and their first-order derivatives are required. In this work, only displacement and traction boundary conditions have been implemented. Displacement boundary conditions are imposed directly in the equations, so the integral version of boundary conditions that is needed, it is restricted to tractions involving only first-order derivatives. Again due to the uncoupling associated with the analog operator, the scalar formulation can be used as starting point.

The construction of the first-order derivatives is based on equation (5.25), valid for collocation points located at "smooth" parts of the boundary and that is repeated for clarity of exposition

$$u_{,s}(\mathbf{z}) + \int_{\Gamma} u(\mathbf{x}) \overset{0}{\underset{=}{Q}}_s(\mathbf{x};\mathbf{z})\, d\Gamma - \int_{\Gamma} q(\mathbf{x}) \overset{0}{\underline{U}}_s(\mathbf{x};\mathbf{z})\, d\Gamma =$$

$$\sum_j \alpha_j \left[\hat{u}_{j,s}(\mathbf{z}) + \int_{\Gamma} \hat{u}_j(\mathbf{x}) \overset{0}{\underset{=}{Q}}_s(\mathbf{x};\mathbf{z})\, d\Gamma - \int_{\Gamma} \hat{q}_j(\mathbf{x}) \overset{0}{\underline{U}}_s(\mathbf{x};\mathbf{z})\, d\Gamma \right]$$

(6.34)

where

$$\overset{0}{\underset{=}{Q}}_s(\mathbf{x};\mathbf{z}) = e_p^s(\mathbf{z}) \overset{0}{Q}_{,p}(\mathbf{x};\mathbf{z}) \qquad \overset{0}{\underline{U}}_s(\mathbf{x};\mathbf{z}) = e_p^s(\mathbf{z}) \overset{0}{U}_{,p}(\mathbf{x};\mathbf{z})$$

(6.35)

whose discretized version is

$$\sum_k^M a_{ik} u_{k,s} + \sum_k^M \overset{\hat{}}{\underset{=}{h}}_{ik,s} u_k - \sum_k^M \underset{=}{g}_{ik,s} q_k =$$

$$\sum_j^R \alpha_j \left[\sum_k^M a_{ik} \hat{u}_{kj,s} + \sum_k^M \overset{\hat{}}{\underset{=}{h}}_{ik,s} \hat{u}_{kj} - \sum_k^M \underset{=}{g}_{ik,s} \hat{q}_{kj} \right]$$

(6.36)

In practice, the use of this equation requires a preliminary calculation, that is common to all operators, and independent of a particular problem. Therefore, this part of the operations can be implemented separately. With this goal in mind and before the construction of the general matrix, coefficients $d_t^p(\mathbf{z})$ defined in (5.21) for all collocation points are calculated and later coefficients $e_p^s(\mathbf{z})$ can be deduced by means of

$$e_p^s(\mathbf{z}) d_t^p(\mathbf{z}) = \delta_{ts}$$

(6.37)

Notice that, in order to obtain these coefficients, only the inverse of the matrix defined by the $d_t^n(\mathbf{z})$ is necessary.

After obtaining these coefficients is immediate to notice that

$$\int_{\Gamma} u(\mathbf{x}) \overset{0}{\underset{=}{Q}}_s(\mathbf{x};\mathbf{z})\, d\Gamma = e_p^s(\mathbf{z}) \int_{\Gamma} u(\mathbf{x}) \overset{0}{Q}_{,p}(\mathbf{x};\mathbf{z})\, d\Gamma$$

(6.38)

And, therefore

$$\hat{\underline{h}}_{ik,s} = e_p^s(\mathbf{z})\,\hat{h}_{ik,p} \qquad (6.39)$$

The next step is to particularize the boundary conditions of the problem, which in this case are pure tractions or displacements. Since displacements conditions are applied directly to the equation, only the traction operator must be analyzed.

In the case of a general material, the traction equation is

$$t_i = \sigma_{ij}n_j = C_{ijkl}u_{k,l}n_j \qquad (6.40)$$

The discretized equation of type **q-BIE** associated with a boundary point located on $\mathbf{z}_i$ (see (4.49)) is

$$\sum_k^M \mathbf{P}_{ik}\mathbf{u}_k - \sum_R^M \mathbf{L}_{ik}\mathbf{q}_k - \sum_j^R \boldsymbol{\alpha}_j\left[\hat{\mathbf{e}}_{ij} + \sum_P^M \mathbf{K}_{ik}\hat{\mathbf{R}}_{kj} - \sum_k^M \mathbf{L}_{ik}\hat{\mathbf{q}}_{kj}\right] = \mathbf{t}_i \qquad (6.41)$$

Comparing equation (6.29) with equation (6.30), and taking into account the expressions of equation (6.36), matrix $\mathbf{P}_{ik}$ is formulated as

$$\begin{bmatrix} C_{1j1l}\hat{\underline{h}}_{ik,l}n_j & C_{1j2l}\hat{\underline{h}}_{ik,l}n_j & C_{1j3l}\hat{\underline{h}}_{ik,l}n_j \\ C_{2j1l}\hat{\underline{h}}_{ik,l}n_j & C_{2j2l}\hat{\underline{h}}_{ik,l}n_j & C_{2j3l}\hat{\underline{h}}_{ik,l}n_j \\ C_{3j1l}\hat{\underline{h}}_{ik,l}n_j & C_{3j2l}\hat{\underline{h}}_{ik,l}n_j & C_{3j3l}\hat{\underline{h}}_{ik,l}n_j \end{bmatrix} \qquad (6.42)$$

So to calculate matrix $\mathbf{P}_{ik}$, the following terms must be obtained

$$\begin{bmatrix} \hat{h}_{ik,1} & \hat{h}_{ik,2} & \hat{h}_{ik,3} \end{bmatrix} \qquad (6.43)$$

where, again, these terms are independent of the differential operator governing the problem.

To obtain the terms $\mathbf{R}_{ik}$ and $\hat{e}_{ij}$ a similar procedure is followed.

If this equation is applied to a collection of M collocation nodes located on the boundary, the discretized version of the **q-BIE** block of equations is obtained, which can be written in matrix form as

$$\mathbf{Pu} - \mathbf{Rq} - \left[\hat{\mathbf{E}} + \mathbf{P}\hat{\mathbf{U}} - \mathbf{R}\hat{\mathbf{Q}}\right]\boldsymbol{\alpha} = \mathbf{t} \qquad (6.44)$$

The sizes of the elements of this equation are

- $\mathbf{u}$, $\mathbf{q}$ dimension $M \cdot d \times 1$

- $\mathbf{P}$, $\mathbf{R}$ dimension $M \cdot d \times M \cdot d$

- $\hat{\mathbf{U}}$, $\hat{\mathbf{Q}}$, $\hat{\mathbf{E}}$ dimension $M \cdot d \times R \cdot d$

- $\boldsymbol{\alpha}$ dimension $R \cdot d \times 1$

- $\boldsymbol{\gamma}$ dimension $M \cdot d \times 1$

6.2.4   System of equations assembly

The set of integral equations defined in the previous section, are combined to assemble the algebraic system of equations able to solve the boundary value problem proposed.

From the numerical point of view, since in this work a **LU** decomposition has been used as solver, the matrix inversion involves most of the computation time, even with a modest number of degrees of freedom. It is therefore imperative to reduce the size of the matrix as much as possible. Therefore, although the unknown tractions can be included directly in the system of equations, it is convenient to carry out its calculation in the postprocessing stage.

The balance equations-unknowns is similar to chapter four. So with this in mind and assuming, without loss of generality that

- A three-dimensional problem

- M collocation and approximation nodes

- Displacement boundary conditions on D nodes

- Traction boundary conditions on $M - D$ nodes

- $R \times 3$ approximation functions of $\hat{b}$

Since, only q-BIE equations have been imposed where the traction is known we have

- $(M - D) \times 3$ unknowns associated with u

- $M \times 3$ unknowns associated with q

- $R \times 3$ unknowns associated with α

And there are

- $M \times 3$ equations **u-BIE**

- $(M - D) \times 3$ equations **q-BIE**

- $R \times 3$ equations **b-BIE**

The result is an algebraic system containing $(2M - D + R) \times 3$ equations and $(2M - D + R) \times 3$ unknowns that can be solved using any standard algorithm.

The final matrix of the system can be defined conceptually as follows.

The vector **u**, which includes all the values of the displacement in the approximation nodes, is divided into two parts: $\mathbf{u}_1$, of dimension $(M - D) \times 3$, which includes the unknown displacements and $\mathbf{u}_2$, of dimension $D \times 3$, which includes known displacements. Similarly matrices **H**, **K** and **P** are separated into two parts. Respectively each part includes the columns multiplying elements of $\mathbf{u}_1$ or $\mathbf{u}_2$. The part multiplying the known displacements $\mathbf{u}_2$ will be moved to the independent term of the equation. Figure (6.1) shows a block diagram containing the composition of the system once assembled.

Figure 6.1: Blocks of the algebraic system of equations

6.2.5 Postproccessing

After solving the system of equations and obtaining the values of displacements u_i, fluxes q_i and coefficients α_j of the approximation of functions, a postprocessing procedure can be performed in order to calculate unknown tractions, internal displacement or internal stresses.

The procedure is completely analogous to the previously presented presented (and to the standard BEM), and therefore has no further interest in it.

As a brief example, traction can be calculated by equation (6.41), applied to all the collocation nodes where the traction value is desired.

In the case of internal displacement, the particularized version of equation (6.21) applied to an internal point is used

$$\mathbf{u}_i = -\sum_k^M \hat{\mathbf{H}}_{ik}\mathbf{u}_k + \sum_k^M \mathbf{G}_{ik}\mathbf{q}_k + \boldsymbol{\alpha}_j \left[\hat{\mathbf{u}}_{ij} + \sum_k^M \hat{\mathbf{H}}_{ik}\hat{\mathbf{u}}_{kj} - \sum_k^M \mathbf{G}_{ik}\hat{\mathbf{q}}_{kj} \right] \quad \boxed{6.46}$$

Finally, internal stresses calculation can be carried out following a similar procedure as used for the deducing the block **q-BIE**. Therefore, from the first-order derivatives of the integral equation, the internal stress equation can be constructed. In this case, the integrals are regular and the formulation results in

$$\boldsymbol{\sigma}_i = -\sum_k^M \mathbf{S}_{ik}\mathbf{u}_k + \sum_k^M \mathbf{T}_{ik}\mathbf{q}_k + \boldsymbol{\alpha}_j \left[\hat{\sigma}_{ij} + \sum_k^M \hat{\mathbf{S}}_{ik}\hat{\mathbf{u}}_{kj} - \sum_k^M \mathbf{T}_{ik}\hat{\mathbf{q}}_{kj} \right] \qquad (6.47)$$

Taking into account that the stress operator is

$$\sigma_{ij} = C_{ijkl}u_{k,l} \qquad (6.48)$$

the block $\mathbf{S}_{ik}$ can be written as

$$\begin{bmatrix} C_{111l}\hat{\underline{\underline{h}}}_{ik,l} & C_{112l}\hat{\underline{\underline{h}}}_{ik,l} & C_{113l}\hat{\underline{\underline{h}}}_{ik,l} \\ C_{121l}\hat{\underline{\underline{h}}}_{ik,l} & C_{122l}\hat{\underline{\underline{h}}}_{ik,l} & C_{123l}\hat{\underline{\underline{h}}}_{ik,l} \\ C_{131l}\hat{\underline{\underline{h}}}_{ik,l} & C_{132l}\hat{\underline{\underline{h}}}_{ik,l} & C_{133l}\hat{\underline{\underline{h}}}_{ik,l} \\ C_{221l}\hat{\underline{\underline{h}}}_{ik,l} & C_{222l}\hat{\underline{\underline{h}}}_{ik,l} & C_{223l}\hat{\underline{\underline{h}}}_{ik,l} \\ C_{231l}\hat{\underline{\underline{h}}}_{ik,l} & C_{232l}\hat{\underline{\underline{h}}}_{ik,l} & C_{233l}\hat{\underline{\underline{h}}}_{ik,l} \\ C_{331l}\hat{\underline{\underline{h}}}_{ik,l} & C_{332l}\hat{\underline{\underline{h}}}_{ik,l} & C_{333l}\hat{\underline{\underline{h}}}_{ik,l} \end{bmatrix} \qquad (6.49)$$

6.3 Other aspects

6.3.1 Shape functions, edges and borderlines

In the previous sections, we have introduced the simplifying assumption that the nodes were collocated on smooth boundary points. Usually, the position of collocation and approximation nodes coincides unless its segregation is convenient. The most obvious case is the presence of an edge (or vertex) in

the boundary to discretize, since the assumption of smoothness is violated and the solid angle must be calculated. In figure (6.2) an example of the case of a square element of nine nodes is shown. In this cases the collocation nodes situated on the edges are moved inside the element.
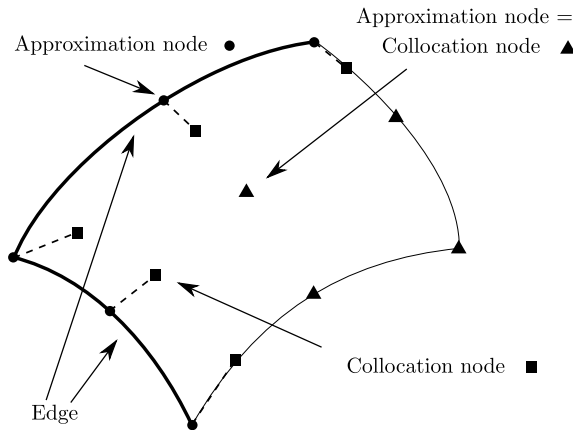
Figure 6.2: Quadratic element with sharp edges

Another aspect that must be taken into account, are the elements that share common edges and collocation nodes. In the case of zero-order elements (constant elements), see figure (6.3), there is a unique approximation node inside the element and, therefore, this case can not occur but in higher order elements, for example second-order elements, is an usual situation. The shape functions used in this work are quadratic, which are C^∞ in the interior of the element, but only C^0 at global level[6] (in particular between elements). This type of interpolation improves the rate of convergence but, due to the introduction of common nodes between adjacent elements, it is necessary to study the behavior of the functions at these nodes.

In the literature, there are several papers that have analyzed the conditions

---

[6]There are several works in the literature that use higher order implementations as [44], or introduce approximations with splines [114], [8].
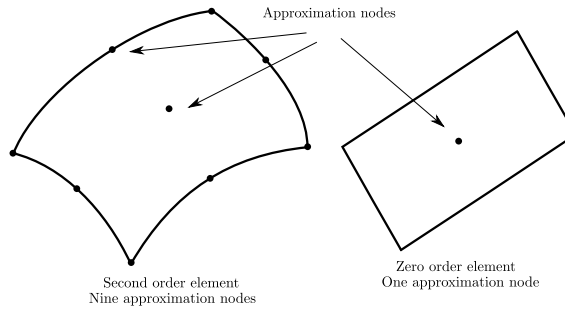
Figure 6.3: Zero and second order elements

for convergence of the Boundary Element Method and the requirements for the approximation functions (e.g. [78], [79] and [47]). Detailed study of this problem is beyond the scope of this work, but it must be emphasized that the lack of C^1 continuity in the approximation functions, introduces errors that will hinder the convergence of the method. As an example, in the analysis of the derivatives of the Boundary Integral Equations associated with the Laplace operator (see A.2), expansions of the variables in terms of their derivatives, which are discontinuous, are introduced.

Additionally it can be noted that, the geometric interpolation[7] also lacks of continuity in the calculation of geometrical derived variables as normal vectors. Considering this, two aspects are potentially problematic. First of all, in the strict sense, free terms associated with angles (or solid angles) between elements appear and second, the discontinuity in the normal vector introduce additional convergence errors[8].

A simple way to avoid these difficulties, is to move all the collocation nodes within the elements. In this case, it is clear that the boundary is smooth

---

[7]Isoparametric approximation is used. Therefore the same functions are used for geometrical and functional approximation.

[8]In a lesser extent, the calculation of the tangential derivative/gradient that appears in the integral equation derivatives is performed by direct derivation of the shape functions. This means that the approximation order of this term have a lesser degree and it contributes to the reduction of the order of convergence.

at the collocation point and, consequently, functions describing both the geometry and the variables are C^∞. The drawback of this approach is the considerably increment of unknowns for the same number of elements. For example, in the case of a square mesh composed of $n \times n$ elements the unknowns are increased from $9n^2$ to $4n^2 + 4n + 1$.

6.3.2 Multidomains

The last aspect to be considered is the multidomain implementation. Three cases are analyzed including inhomogeneous domains and combinations of homogeneous domains (for which the standard boundary elements algorithms are used) with inhomogeneous domains.

The case of combination of domains is divided into two cases, which are treated differently because the interface conditions vary. This depends on whether there is discrete gap of properties at the interface or not. In the absence of discrete gap of properties, it is clear that the problem could be solved without interface applying the algorithm AEM-BEM to the entire domain. The use of multidomains in this case, arises to improve numerical stability and accuracy of results (within the framework of BEM can be found in the literature several works in this line, i.e. [42] o [84]).

A brief analysis of the problem, focus on the coupling equations of the interfaces is performed . Let assume, without loss of generality, a problem as shown in figure (6.4), where we have a solid with different domains and mixed boundary conditions. Classic perfect adhesion conditions apply at the interfaces. Thus the interface compatibility equations are

$$\begin{aligned} \mathbf{u}^{ij} &= \mathbf{u}^{ji} &\quad \text{on} \quad \Gamma^{ij} \\ \mathbf{t}^{ij} &= -\mathbf{t}^{ji} &\quad \text{on} \quad \Gamma^{ij} \end{aligned} \tag{6.50}$$

Without any further assumption about the properties of each subdomain and taking into account the aspects previously remarked, standard BEM schemes are applied for homogeneous subdomains and AEM-BEM schemes
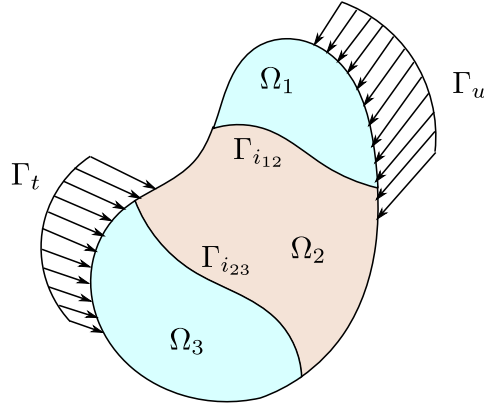
Figure 6.4: Multidomain consisting in three subdomains

(detailed in this work) for inhomogeneous subdomains. These equations are applied to the external boundaries of the problem (and in the case of an inhomogeneous domain to a collection of internal points).

The additional compatibility equations depend on the properties of the associated subdomain and are applied in the interfaces.

Inhomogeneous-Inhomogeneous interface without gap of properties

In this case, the coupling equations can be replaced by

$$
\begin{array}{lll}
\mathbf{u}^{ij} = \mathbf{u}^{ji} & \text{on} & \Gamma^{ij} \\
\mathbf{q}^{ij} = -\mathbf{q}^{ji} & \text{on} & \Gamma^{ij}
\end{array}
\tag{6.51}
$$

Intuitively, the equal displacement conditions in the interface, also implies an equal tangential derivative of the displacement. The imposition of equal traction value condition with same properties, clearly implies that the normal derivatives (fluxes) must also be equal (opposite).

The balance of equations and unknowns in each collocation point is performed. We have three displacements, three fluxes and three tractions in each side of the interface for a total of 18 unknowns. The use of the 6 compatibility equations (6.51) and the addition of 6 **u-BIE** equations from each side of the interface results in 12 equations involving 12 unknown displacements and fluxes .

In this case, the equations related to tractions at the interface are not required and the tractions on the boundary can be calculated in the post-processing stage.

In practice, the compatibility equations are introduced directly into the system, so that only **u-BIE** type equations are used in the interface in order to solve the problem.

Inhomogeneous-Inhomogeneous interface with gap of properties

In this case, the compatibility equations can not be replaced, since the difference of properties means that the fluxes are different.

Therefore, the 18 unknowns mentioned above appear. The use of the 6 compatibility equations (6.50), along with 12 **u-BIE** and **q-BIE** equations from each side of the interface results in a determined system of equations.

Noticed again that the compatibility equations are entered directly into the system, so that only type **u-BIE** and **q-BIE** equations are used in the interface in order to solve the problem.

Inhomogeneous-homogeneous interface

This case involves the coupling of different algorithms and requires more careful analysis. In the homogeneous part standard BEM algorithm is used, so the equation in matrix form can be written as

$$\mathbf{Hu} - \mathbf{Gt} = \mathbf{Sb} \qquad \boxed{6.52}$$

where it has been assumed the existence of body forces, which have been treated by dual reciprocity techniques, as discussed previously (see section 2 of chapter 3).

The balance of equations and unknowns in each collocation point is performed again. We have twelve unknowns related to displacements and tractions and three more unknowns related to the fluxes of the inhomogeneous side of the interface. In this case the 6 compatibility equations (6.50) are combined with 6 **u-BIE** and **q-BIE** equations from the inhomogeneous side of the interface and 3 **u-BIE** (standard BEM) equations from the homogeneous side of the interface.

It should be noticedd that the traction unknowns are associated with collocation nodes (the traction is associated with the coordinate **z**) for the AEM-BEM scheme applied in the inhomogeneous part. In the other hand the tractions unknowns are associated with approximation nodes for the standard BEM applied in the homogeneous part. This implies that if the collocation nodes and the approximation nodes do not coincide, the system is not determined without further operations.

In these cases, it is necessary to relate the value of tractions in the collocation points with the value of the traction in the approximation points. This coupling can be done by means of the shape functions.

Generally, in an element, we can write

$$\mathbf{t}^c = \mathbf{\Phi}^c \mathbf{t}^a \qquad \boxed{6.53}$$

where $\mathbf{t}^c/\mathbf{t}^a$, is a vector containing the values of the traction at the collocation/approximation nodes and $\mathbf{\Phi}^c$ is a matrix containing the values of the shape functions at the collocation nodes.

Inverting this equation, we can replace the variables $\mathbf{t}^a$ of the homogeneous part for a liner combination of $\mathbf{t}^c$ in the case that the nodes do not coincide, and therefore complete the system of equations.

7

Object-Oriented Programming in BEM: A FORTRAN 2003 implementation

Object Oriented Programming has tremendous potential in the scientific software field (see, for example, [49]. [16] or [72]) This approach to programming produces easily maintainable, reusable and scalable computational codes. This is especially critical if we consider that, one of the main unintended consequences of code optimization in order to minimize the execution time of each run, was frequently unstructured, unreadable and often unreliable code, which even the program's author had difficulty in debugging.

As it has been m in previous chapters the Boundary Element Method (see for example [118], [5]) has been used successfully for solving many engineering problems governed by systems of partial differential equations. As the steady march of progress in individual scientific disciplines increases BEM techniques field of application, the need of integrated and scalable codes is more and more clear. Most computers programs for BEM are written in FORTRAN language (see examples in [14] or [39]), not only in the last 90/95 revision but even in the 77 version. Although former FORTRAN revisions (prior to 2003) have not native support for Object Oriented Programming most of the key features can be emulated efficiently in the 90/95 revision [2], [46]. Despite these works there is only a limited number of works involving object oriented programming in FORTRAN

(for example [106]) available in the literature. Most of the works involving object oriented programming applied to numerical methods are focused on Object Oriented Analysis (like [74]) or `C++` Implementations (like [66], [117] or [96]). The release of the last FORTRAN 2003 revision (see further details in [80]) fully supporting Object Oriented Programming (OOP) permits us to transfer all this methodology directly to native FORTRAN without elaborated construction to emulate Object Oriented features. Inheritance, Polymorphism, dynamic type allocation, encapsulation can be used directly. In this work an approach of object oriented programming in Boundary Element Method is used to design the core of a modular program able to easily scale and integrate different techniques. The Object Oriented analysis and design is discussed in detail in order to demonstrate the advantages of this approach.

7.1 System requirements and general considerations

First of all we need to define the requirements of our code.

- Multi-Domain

- Multi-Physics: several kinds of problems supported including 2D/3D

- Able to use different integration schemes at element level

- Able to use different BEM algorithms

- Extensive use of dynamic memory (writes data once / reads many)

- Simultaneous use of several interpolations, elements

- Support different kinds of boundary conditions including inter-phases

- Reduction of the computational cost

The implementation of the previous requirements in FORTRAN 2003 often demands the design of structures adapted to the peculiarities of this language. Although the design of software to validate the algorithm has consumed

Figure 7.1: Geometrical entities

most of the time dedicated to this work, the implementation details strictly related to the programming language will be mostly ignored because they are not intrinsically related to the thesis aim.

Two examples of implementation aspects are exposed. The simultaneous use of multiple types of elements in one array is explained. FORTRAN 2003 does not directly allow arrays consisting in elements of a class and its extensions. To overcome this difficult we can use the following approach. We can create a derived-type which has a polymorphic variable as component. In order to implement the linked structure depicted in the figure (7.1), derived types with pointers as components are defined. Additionally, a constructor is created and added to the class to allow the allocatable components to be type defined with a method of the class. This constructor can have, as "intent in" arguments, labels or directly data to be analyzed to decide the proper type allocation. Figures (7.2) and (7.3) show code excerpts with the main ideas exposed.

```
type       :: selement
           class(element),allocatable      :: n
end type selement

type       :: p_selement
           type(selement),pointer          :: n
end type p_selement
```

Figure 7.2: Derived Type with polymorphic component

```
type       :: construc
    contains
    procedure,nopass     :: new => selecelemento
end type construc

type       :: elemento
    type(construc)       :: constructor
    type(p_snodo),allocatable,private:: nodog(:)
    contains
. . . .
end type element
```

Figure 7.3: Derived Type and its constructor

Another inconvenience is that many times a type is defined primarily to serve
as a base type for extension. In this situation methods have not a natural
implementation. Although FORTRAN provides us deferred procedures and
abstract types that is not an efficient solutions many times. If we have, for
example, several kinds of properties for different types of regions, we can
simply create a method for every class with different types of arguments
and use interfaces to call those methods with the same order. If deferred
procedures in the top class are used that will force us to create all the
methods in every extension. To avoid this situation we use dummy functions
in the top class that theoretically will never be invoked. Figure (7.4) shows
an example of a dummy function. An error message is included to inform
us at what point is the allocation problem, if it exists.

```
subroutine innodec0 (element1,pnode1)
     implicit none
     class(element),intent(inout) :: element1
     type(p_snode),intent(in)     :: pnode1(:)
     print*,'Module element, subroutine innodec0'
     print*,'Dummy function'
     stop
     ...
end subroutine innodec0
```

Figure 7.4: Dummy function example

The typical multi–region problem discretized according to the BEM and solved using the corresponding algorithm has the following basic steps presented in Table (7.1). Although some variations need to be introduced in more sophisticated methods the essential steps remain generally the same[1].

STEP 1 Read Data
STEP 2 Create system of equations
STEP 2.1 For each region
STEP 2.2 For each Element apply boundary conditions
STEP 2.3 For each collocation node evaluate integrals and set the results in the matrix
STEP 3 Solve the system
STEP 4 Introduce the solution in the approximation nodes
STEP 5 Compute additional values

Table 7.1: Basic steps in a BEM algorithm

One of the most significant issue of almost every BEM problem is that the kernels of the involved integrals are singular in the collocation points and only exist in the Cauchy principal value or Hadamard finite parts sense. Several

---

[1]Some special variations of BEM like Fast Multipole [71] or Adaptive Cross Approximation [13] among others need more deep modifications and they are not initially included in this approach.

techniques have been used to solve this topic like direct analytical integration, indirect approach (see for example [19]) or regularization methods (for example [35]). In terms of implementation the key point is that integration methods are problem and element type dependent so an integration library will be mandatory in general BEM software. In order to cover a general implementation our design needs to be flexible and able to incorporate additional equations that arise whenever derivatives of the previous boundary integral equation are taken not only to evaluate those derivatives as part of some post-process procedure but as part of a solution algorithm.

So two levels of dependency have been identified. The first one is related with the kernel integration and generally is related with the type of problem and the type of element. Eventually a third agent can be involved if several options are available: the user. Then the integral methods are set at element level during program execution. The second one includes extensions of classic BEM procedures (like AEM-BEM) that introduce new equations so the general algorithm is problem dependent. Normally these extensions are related to special regions (for example non-homogeneous) so the general algorithm is set at region level during program execution.

7.2 Program Organization

The object oriented design proposed herein is divided at top level in two major categories as showed in figure (7.5): Geometrical objects and libraries. The first one is the direct translation to classes of the real objects that we manage when we define the discretized geometry. These objects are organized in arrays and connected using pointers capturing the real connection among them. The second one includes groups of routines related with different type of problems, integration methods, boundary conditions or general purpose routines shared among multiple problems.

A brief explanation of every category is included. The objective is to give a general idea of the different modules without going into detail.

(a) Geometrical entities   (b) Libraries

Figure 7.5: Main Program Categories

7.2.1 Geometrical entities

Class Point

It implements a geometric point in a $\mathbb{R}^n$ space. An object point has simply an allocatable array of real coordinates covering problems in different dimensions. Several entities can contain the same point so the point class must be a separated object.

Class Node

This is the class that implements a general node. Boundary Element Method needs, in its general case, three basic types of nodes with different functionalities that may have different positions in the space.

Geometrical nodes simply contain a pointer to a point that situate it in the space and is used to define the geometry of the element. The *collocation nodes* are used as source point of the fundamental solution and therefore is where we particularize our integral equations. They contain a pointer to a
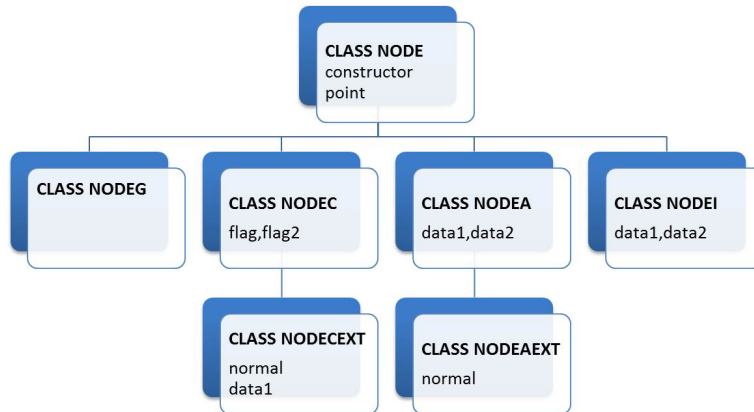
Figure 7.6: Class node and extensions

point and several logical flags that are used during the integration process. The last type is the *approximation nodes*. They contain again a pointer to a point and two general polymorphic fields which contain the physical variables of the problem.

Additionally we have internal nodes with the same components as approximation nodes, extended approximation node with a field to contain the local normal to avoid recalculation and extended collocation node with a field to contain the local normal and one general polymorphic field which contain physical variable of the problem. In figure (7.6) the class node and its extensions are represented.

Several procedures are available to create nodes, insert points, retrieve coordinates, allocate, insert and read data...

Class Element

This is the class that implements a general boundary element. The basic class only contains pointers to the geometric nodes. Extension of this class includes elements with approximation nodes and collocation nodes and their position in local coordinates. Several extensions are implemented, for example, two-dimensional quadratic elements and 3D quadratic elements based on 6-nodes triangles or 9-nodes squares. In this implementation of the direct BEM the evaluation of the matrices is done at element level, but integration is not essentially coupled with the element. For this, the element has an allocatable class integrator whose purpose is to alleviate elements objects from integration tasks. This component of the element can be changed on run time.

Several procedures are available to create elements, insert nodes, retrieve nodes, geometrical properties...

Class Boundary

This is the class that implements a patch of hte boundary. This class has been designed with two main objectives. The first one is to capture the natural relation between a boundary and its boundary condition. Every boundary has, besides pointers to elements, an allocatable class boundary condition whose purpose is to perform all the calculations related to the different types of boundary conditions implemented. The formulation of the system of differential equations give us boundary conditions normally in terms of faces or part of the surface so it is logical to keep this entity in our design. The second one is to capture the geometrical "faces" of the domain where the normal vector is continuous to avoid problems with calculations of solids angles in cases that we have corners or edges between boundaries. This is a typical situation that normally requires displacing the position of the collocation node or calculating the solid angle of the node between faces. It is easier and more generic to simply implement an offset.

A consequence of this is the fact that these boundaries can be treated as isolated entities in terms of collocation and approximation nodes. This means that in our algebraic system of equations we can identify regions not only related to collocation or approximation nodes but related to boundaries (in terms of their collocation and approximation nodes) so the assembly of the matrix is independent at boundary level. After the construction of a boundary a procedure is run to create an internal map of nodes to assembly the system.

Class Region

This is the class that implements a region. Basically an object region contains pointers to boundaries, and the direction of these boundaries. The extensions of this class refer to homogeneous or non-homogeneous and they include data related with its material properties. Additionally it can contain pointer to inner nodes that can be used whether to calculate the value of the physical variable at them or as base to approximation functions like in Dual Reciprocity Methods. There are two relevant procedures in a region object. One is the high level integration algorithm. Properly the integration is carried out at element level but is at region level where the most relevant part of the algorithm organization is reflected. At this level is where we call procedures that select the proper integration method, and the procedures that implements boundary conditions in the approximation nodes. Additionally, at this level, we directly assembly the final matrix of equations with data obtained from the boundary conditions.

The second relevant procedure carried out is the introduction of the solution of the system of equations in the approximation nodes. Strictly this procedure only passes the relevant data to the boundary condition which performs the task.

Class Multidomain

This is the class that implements a multidomain. Strictly every problem is multidomain but sometimes there is only one domain. This class contains pointers to regions as data. Additionally it creates an internal map[2] which relates boundaries with matrix parts and it permits to assembly the matrices correctly. Its main procedure is the one that creates the matrix of the system of equations. At this level the global matrix is created and the proper parts are passed as pointers to the regions to fill them. Additionally a postprocess routine is implemented to calculate any additional value.

7.2.2 Libraries

Problem Library

This Library consists of two classes of objects. The class problem, at this level, consists basically in collection of useful labels, like the kind and dimension of variables involved, that need to be used during execution. It has been preferred to separate problems from fundamental solutions not only for sake of clarity but some fundamental solutions are shared among several problems and some problems needs several functions. Two and three dimensional Laplace problems, two and three dimensional elastosatic in homogeneous domains and three dimensional elastostatic in non-homogeneous domains problems are implemented at this moment.

The second class is the fundamental solution class. Their procedures, apart from construction and initialization constants and source points, are basically related with their evaluations and of their derivatives, and any related function that can be requested by an integration algorithm.

---

[2]More auxiliary variables are stored in every domain to avoid recalculations

Boundary Condition Library

This Library consists in the class boundary conditions and its extensions. The class boundary conditions implements different types of boundary conditions including inter-phases. Three main classes have been implemented.

- Basic boundary conditions where simply a function give us directly the value of some unknowns. Extensions have been constructed for standard BEM algorithm and AEM-BEM techniques.

- General linear boundary conditions where the unknowns are related through a general system of equations with variable coefficients.

- Inter-phase with perfect contact condition. This boundary condition implements compatibility and equilibrium equations directly over the interfaces during the construction without performing additional operations at the end of the matrix assembly. Three inter-phases are implemented. Homogeneous-Homogeneous and Non-homogeneous - Non-homogeneous without jump of properties, Homogeneous - Non-homogeneous and Non-homogeneous - Non-homogeneous with jump of properties.

The object boundary condition has two main procedures. The first one is responsible of introducing the boundary conditions in the approximation nodes and calculate the final positions and multipliers that we need to use in the matrix of equations. The second one is responsible of introducing the solution of the system in the approximation node.

As we can see, this kind of object is closely linked with boundaries objects. In fact boundary conditions "are part of" a boundary through a pointer but the same boundary condition can be part of several boundaries.

Integral Library

This library consists of several classes that carry out the integration at element level. It is aforementioned that in a generic BEM program the integration algorithm is problem/element dependent especially due to the singular character of the fundamental solution. Mainly three classes are gathered in this library.

The first class is the quadrature class. It implements all the objects that we need to perform numerical integrations through Gaussian quadratures. Basically it has procedures to obtain the weights and position involved in these numerical integrations.

The second class is the interpolation class. It has been designed to perform the entire task related to interpolate values. Obviously is closely linked to the element class and, in fact, all the calculations to obtain coordinates, jacobian, normal and other values inside the elements are performed by this class. Additionally, it performs all the interpolation of variables that can be, in a general case, different from the geometrical interpolation.

The third one is the integrator class. It implements all the integration algorithms involved in the boundary element method. In the literature three main types of integrals can be found: regular, near-singular and singular (and eventually hypersingular and so on). Regular and near-singular integrals can be implemented in a quite general way with allocatable arrays which shapes are set in run time, but singular integrals are strongly problem/element dependent. Basically, the object integrator performs three kinds of integrals. Suitable subroutines are selected with the constructor of the object as part of an element, depending on three types of input data as is showed in figure (7.7): type of problem, type of element and user inputs. The user input is designed, for example, to pass the number of gauss point to perform integration, or labels to select a specific integration procedure. This user input is optional and the constructor has default options to select the appropriate set of methods. After the integrator is initialized a condition parameter with the level of singularity is passed when the integrate procedure
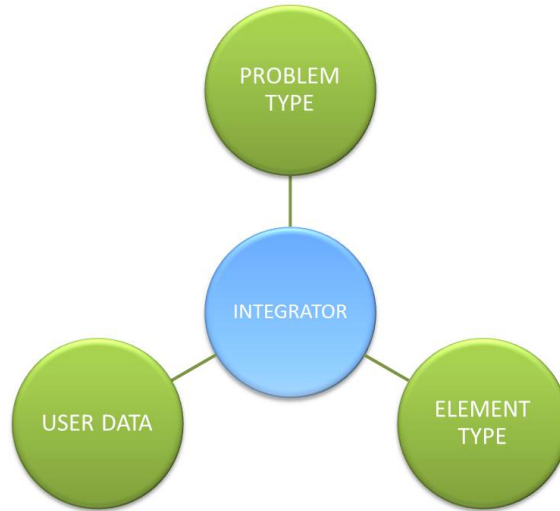
Figure 7.7: Selection of integrator

is called to select the appropriate method.

An important issue to highlight is that all the process outlined in table (7.1) fixed an element and cover all the collocations nodes. This fact combined with a constant number of gauss point permits us to save inside subroutines a great part of our calculations. Following this idea we only calculate once variables like positions, normals, jacobians ... and store them using "save" command achieving execution time reductions of 30-40%.

General Math Library

This library contains additional classes used for several tasks related with mathematical operations. Briefly, class data is designed as a container to store complex or real data in arrays. This class is used to store the values of the physical variables and the matrices of the system. This level of abstraction ensures us that the program can be upgraded naturally to

```
subroutine addsomething (list1,data,number)
              class(listg),intent(inout)     :: list1
              class(*),intent(in)            :: data
              . . . . .
              call list1%current_p%in(data)
              . . . . .
end subroutine addsomething

subroutine getsomething (list1,data,number)
              class(listg),intent(in)        :: list1
              class(*),intent(out)           :: data
              . . . . .
              call list1%current_p%out(data)
              . . . . .
end subroutine getsomething
```

Figure 7.8: Unlimited polymorphic class in linked list

dynamic problems in the frequency domain. It contains procedures to introduce or read data in specific positions.

Class radial basis function is designed in a similar way as fundamental solution class and is used to perform every task related with evaluations of these kind of functions and their derivatives. This class is intended to be used in problems with dual reciprocity methods involved.

Auxiliary Library

The last library has been divided from the previous library to gather the classes not specifically related with the Boundary Element Method.

The class generic list is designed as linked list able to store all type of data. The new FORTRAN standard includes an unlimited polymorphic class that can be used to achieve that objective as showed in figure (7.8).

The class lecture is designed simply to read data from a text file used to define the geometry, problem, boundary conditions and in general all the data needed to define the problem.

7.2.3   Conclusions

This chapter presented a brief study of the new supported features in FORTRAN 2003 and its use to design a global BEM program in an object oriented style able to easily scale and integrate different techniques. The main classes of the proposed architecture are outlined and their principal methods explained. Some codes excerpts are attached showing relevant aspects of specific topics. The proposed architecture capture the natural way of manipulating geometries and problems parameters, linking objects as we do in theoretical approaches. The procedures to assembly the matrices of the system equations has been designed to optimize execution time to avoid duplication of calculations. At this point of implementation several kinds of problems as Poisson or elastic are fully functional in 3-dimensional geometries with quadratic elements. The addition of new problems only requires the study of particularities of the singular integrations. If there is no valid integrator procedure in the program able to calculate the integrals of the new problem, the addition of a new singular integration method and the corresponding fundamental solution will be the only necessary tasks. Furthermore, the encapsulation of data ensures that the addition of new problems will not affect previous implementation and allow us to split the future work in global solution strategies which are needed to fracture problems or local integration problems to add new singular or near singular integrators.

Additionally the new FORTRAN 2003 standard has showed that every relevant object oriented feature has been successfully implemented. The new object oriented features including polymorphism, encapsulation, dynamic binding, and data hiding, etc... prove that FORTRAN is able to produce fully Object Oriented Programs maintaining backwards compatibility with previous FORTRAN codes preserving previous algorithm as fully functional procedures.

Part III

Results

8

Numerical Examples

In this chapter a series of numerical examples, designed to show the convergence[1] and stability of the methodology shown analytically in the previous chapters are presented.

The order of the examples represents, almost chronologically, different milestones of the numerical implementation. Although formally is not included in the scope of this thesis, an example of the initial developments of this implementation is included. This first example studies scalar second order elliptic operators with variable coefficients and boundary conditions formulated in terms of the variable and its flux defined in a two-dimensional space.

8.1 Diffusion problem

The first numerical example studies the thermal diffusion problem in a plane blade. Mathematically the problem is defined as

$$\nabla \cdot [K(\mathbf{x}) \nabla T] = 0 \quad \text{en} \quad \Omega \tag{8.1}$$

---

[1]It should be emphasized that the convergence of the Boundary Element Collocation Method has not been demonstrated analytically.

where the thermal conductivity $K(\mathbf{x})$ varies according to the law

$$K(\mathbf{x}) = (2x + y + 2)^2 \qquad \boxed{8.2}$$

In figure (8.1) the coefficient K variation along the blade is shown. This problem has been chosen, firstly, because it has analytical solution whose formula is

$$T(\mathbf{x}) = 100 + \frac{6x^2 - 6y^2 + 20xy + 30}{2x + y + 2} \qquad \boxed{8.3}$$

so the error analysis can be done exactly and, secondly, because by a change of variable of type $T(\mathbf{x}) = \dfrac{\tau(\mathbf{x})}{K^{0.5}(\mathbf{x})}$ it is possible to transform the original equation into the Laplace equation.

Taking this into account, a problem with a known fundamental solution is obtained and, therefore, it can be solved directly using the standard Boundary Element Method.

Figure 8.1: Coefficient K variation along the blade

This allow us to have a reference to compare the level of accuracy and the convergence rate of the method. Both methods have been implemented using constant elements[2]. The boundary conditions and geometry of the blade can be found in figure (8.2). Flux boundary conditions are prescribed

---

[2]The implementation was done using Matlab as part of the MSc research prior to this thesis.

along two boundaries and the temperature is prescribed on the remaining four.

The first graph of results (8.3) represents, in double logarithmic scale, the error, according to L_2 norm, of the temperature at boundary points versus the total amount of degrees of freedom of the problem

The L_2 norm is defined as:

$$e_{L_2} = \frac{\|\boldsymbol{d}_{\mathrm{ex}} - \boldsymbol{d}_{\mathrm{num}}\|_{L_2}}{\|\boldsymbol{d}_{\mathrm{ex}}\|_{L_2}} \quad \boxed{8.4}$$

where $\boldsymbol{d}_{\mathrm{ex}}$ is the vector that contains the exact values of the unknowns, $\boldsymbol{d}_{\mathrm{num}}$ is the vector that contains the values of the unknowns calculated numerically and $\|\boldsymbol{d}_{\mathrm{ex}}\|_{L_2}$ and $\|\boldsymbol{d}_{\mathrm{ex}} - \boldsymbol{d}_{\mathrm{num}}\|_{L_2}$ are defined respectively as

Figure 8.2: Geometry and Boundary Conditions

$$\|\boldsymbol{d}_{\mathrm{ex}}\|_{L_2} = \sqrt{\sum_i d_{\mathrm{ex}_i}^2}$$

$$\|\boldsymbol{d}_{\mathrm{ex}} - \boldsymbol{d}_{\mathrm{num}}\|_{L_2} = \sqrt{\sum_i (d_{\mathrm{ex}} - d_{\mathrm{num}})_i^2} \quad \boxed{8.5}$$

In this first example, the approximation function used is the ATPS 2D, which combines radial functions with polynomial terms. Every curve is associated with a fixed distribution of internal nodes and an increasing amount of boundary elements.

Furthermore, the error curve associated with the problem transformed by change of variable and solved using standard BEM is added. In this case, the number of degrees of freedom is equal to the number of elements.

Figure 8.3: L_2 error of the temperature on the boundary

Analyzing figure (8.3) several conclusions can be extracted immediately. It can be observed that each curve has two asymptotes, so that, in general, the curves have a structure as shown in figure (8.4). For a curve with a constant amount of internal nodes, the horizontal asymptote, intuitively, represents the fact that we are not really solving the original problem, but an approximation whose independent term is represented by the accuracy of the radial approximation.

Thus, for a number of internal nodes the problem converges to a solution that it is not the exact one and, consequently, the error has a minimum. The oblique asymptote is associated with the accuracy of the Boundary Element Method. This can be checked clearly by observing the reference curve of the problem solved by change of variables, which has a similar gradient.

The behavior of the error is due in part to the calculation of the integrals,
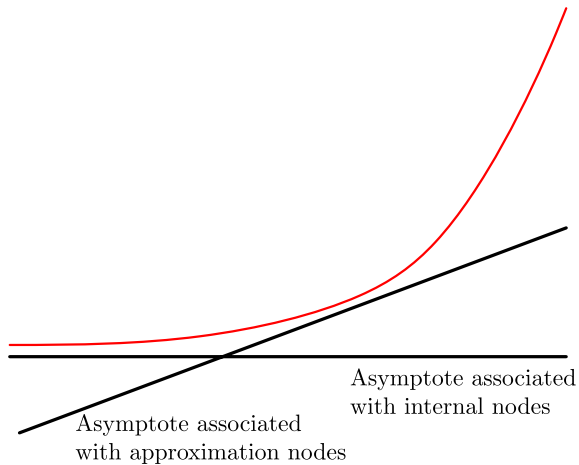
Figure 8.4: Asymptotes of the error curves

which is determined by the discretization of the boundary. In an initial phase, the improvement of the discretization affects both the calculation of the domain integrals (transferred to the boundary), and the original boundary integrals. As the amount of boundary elements grows, the ATPS approximation is good enough, so that the dominant error is associated with the discretization of the boundary. At one point the dominant error returns to be associated with the independent term approximation whereby and it tends to a nonzero value.

In general, it is observed that the error has a similar order of magnitude to the resolution by standard boundary element method (in fact the envelope of the error curves presents lower errors) and that an optimum range of boundary elements for each number of internal nodes exists. According to the tests that have been performed, as a reference value, this optimum range can be estimated between 30 and 50 percent of degrees of freedom associated with the approximation of the independent term. Increasing this ratio does not provide substantial improvements and may even slightly increase the error.
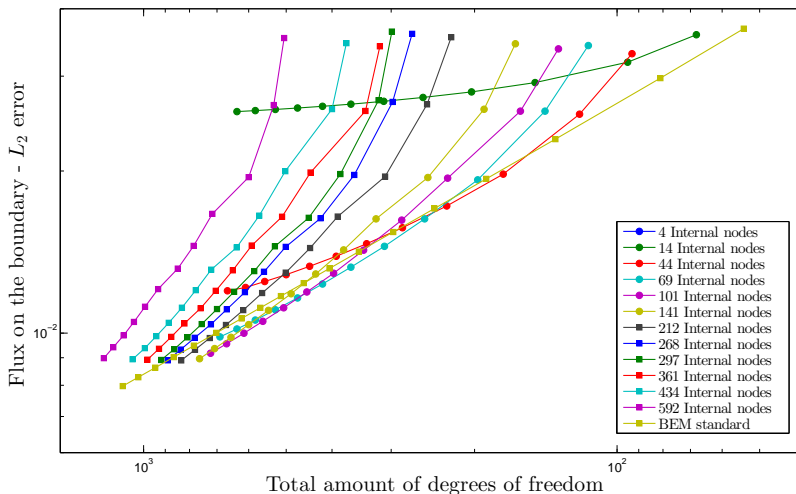
Figure 8.5: L_2 error of the flux on the boundary

The curves representing the error of the flux on the boundary are shown in figure (8.5). In these curves is observed again, the same behavior but some remarks must be done. The error is higher due to the so-called "corner" problem. Nodes adjacent to the areas with jumps in the normal vector have high errors but outside these small areas accuracy is very high.

The results of the next part of the test is shown in figure (8.6), where errors associated with temperature in internal points are represented. Similar behavior in all curves can be observed.

The final analysis performed for this example is a comparative analysis of different types of approximations functions of the term $\hat{b}$. Specifically, the following approximation functions are compared
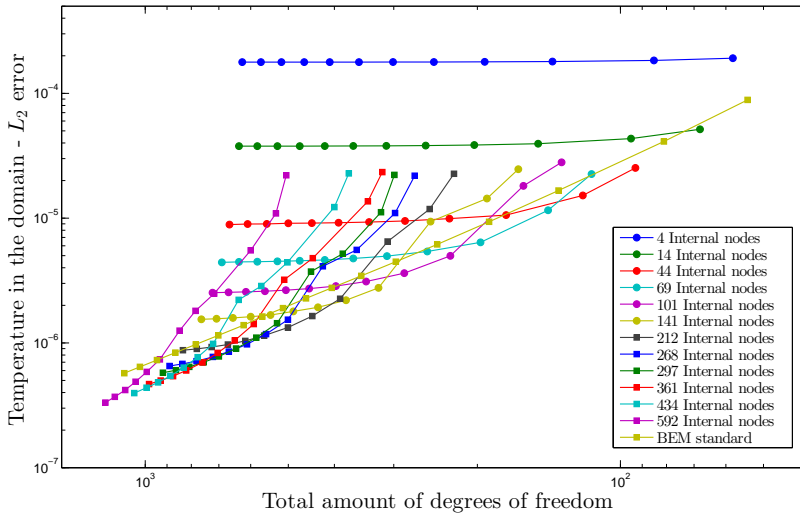
Figure 8.6: L_2 error of the temperature in the domain

- ATPS 2D

- ATPS 3D

- Multiquadrics

- $c + r$

- Spline with several support values

In figures (8.7), (8.8) and (8.9) the error envelopes are shown according to L_2 norm. The temperature and its flux on the boundary and the temperature in the domain are compared for different approximation functions.

The analysis of these curves shows visually that the envelopes have the same rate of convergence than the standard BEM, as expected. It is also noted that the accuracy, to equal amount of degrees of freedom, is better with the AEM-BEM algorithm for several types of approximations.
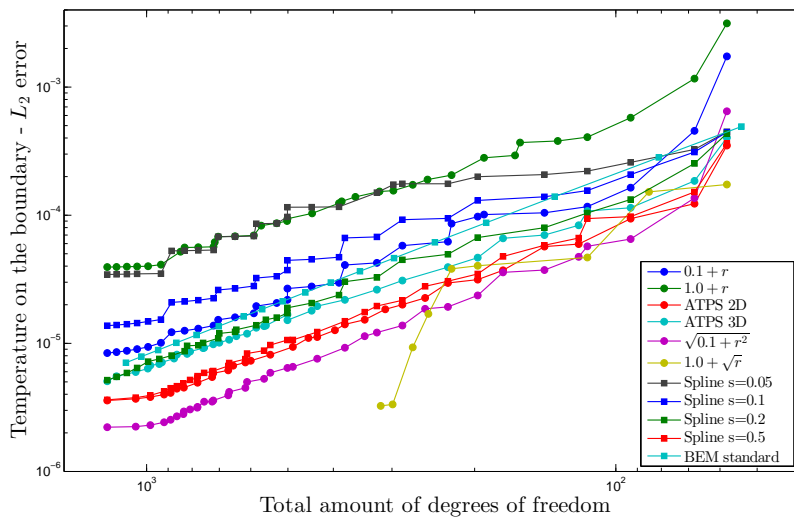
Figure 8.7: L_2 error envelopes of temperature on the boundary
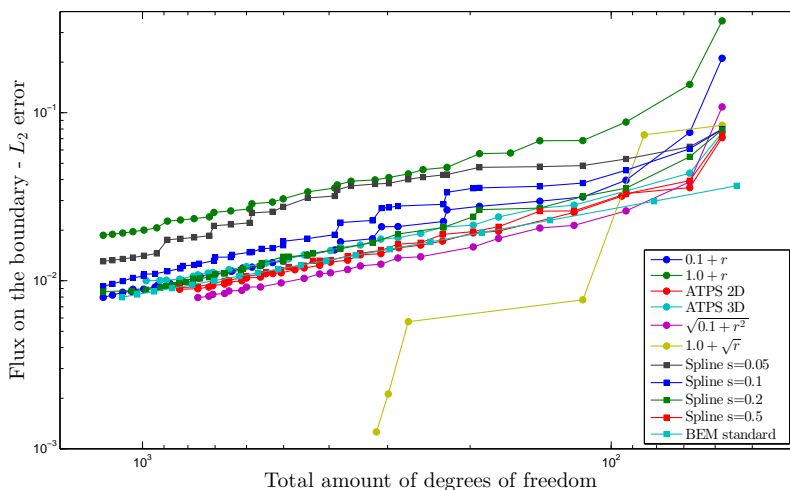
Figure 8.8: L_2 error envelopes of flux on the boundary
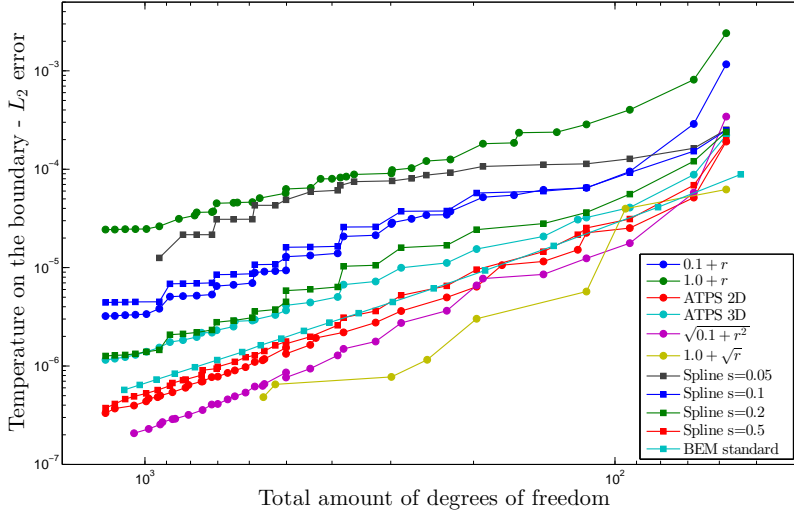
Figure 8.9: L_2 error envelopes of temperature in the domain

This methodology has been used to solve more examples with equations governed by more complex operators. The results are similar and are not included in this work for the sake of brevity it. The main conclusions are:

- Linear approximation families of type $f(r) = c + r$ show the worst overall performance.

- In approximation families with supports or parameters, the accuracy of the results is strongly influenced by the value of these constants.

- Approximations with polynomial terms (ATPS and Splines) have shown no problems of numerical stability, but the conditioning of the resultant matrices using multiquadrics functions is highly dependent on the parameter c, and the result can vary from the lower error in the solution to diverge due to bad conditioning of the matrix.

- The ATPS 2D approximation functions have the best balance of

accuracy, stability and simplicity. In addition no further considerations (support or parameter values) are needed for its definition.

8.2 Three-dimensional elastostatic problem

8.2.1 Dirichlet boundary conditions

This example is a slightly modified version of the one included in the article (see [98]). In that paper the AEM-BEM methodology, analyzed in this thesis, is presented including its implementation in 3D using Dirichlet boundary conditions.

In order to evaluate the error accurately, a problem with known exact solutions have been generated for different geometries. In this case, the displacement field and the properties of the domain are defined. With these data, it is simple to obtain the associated stresses and body forces that allow to define completely the problem. Also, fluxes (which, in this case, are variables without physical meaning), internal stresses or any desired field can be obtained easily.

The displacement in the domain and on the boundary are given by,

$$u_x = \sin x; \qquad u_y = \cos y; \qquad u_z = e^z \qquad \boxed{8.6}$$

and the material properties vary according to the equations

$$\lambda = 100{,}000 - 50{,}000(x - y + z); \quad \mu = 80{,}000 + 80{,}000(x + y - z) \qquad \boxed{8.7}$$

Geometry 1

A simple rectangular prism (see figure 8.10) is used in the first example to test the capabilities of the algorithm to deal with inhomogeneous problems. In

Figure 8.10: Geometry and boundary discretization of a rectangular prism

this case the dimensions of the prism are $X \times Y \times Z = 0.20 \times 1.00 \times 0.20$. The geometry has been selected to allow a regular refinement in every coordinate of the boundary element mesh and the internal nodes distribution.

Taking into account the dimensions of the domain, note that the coefficient λ varies from a minimum of 55,000 at $(0.20, 0.00, 0.20)$ to a maximum of 150,000 at $(0.00, 1.00, 0.00)$, and coefficient μ from minimum of 64,000 at $(0.00, 0.00, 0.20)$ to a maximum of 176,000 at $(0.30, 1.00, 0.00)$. The material is, therefore, strongly inhomogeneous, and the Lame constants vary independently. The Poisson ratio ν is also not constant and varies from a minimum of 0.2215 at $(0.20, 1.00, 0.00)$ to a maximum of 0.2922 at $(0.00, .00, 0.20)$.

The problem is solved, at this stage, with ATPS 3D as approximation function, and the results are presented identically to the previous section by means of several graphs representing the L_2 error distribution versus the amount of degrees of freedom. Three graphs are provided to analyze the convergence and stability of the algorithm. Since all boundary conditions are given in displacements, graphs do not show the error of this variable. The values of the flux on the boundary are calculated at approximation nodes
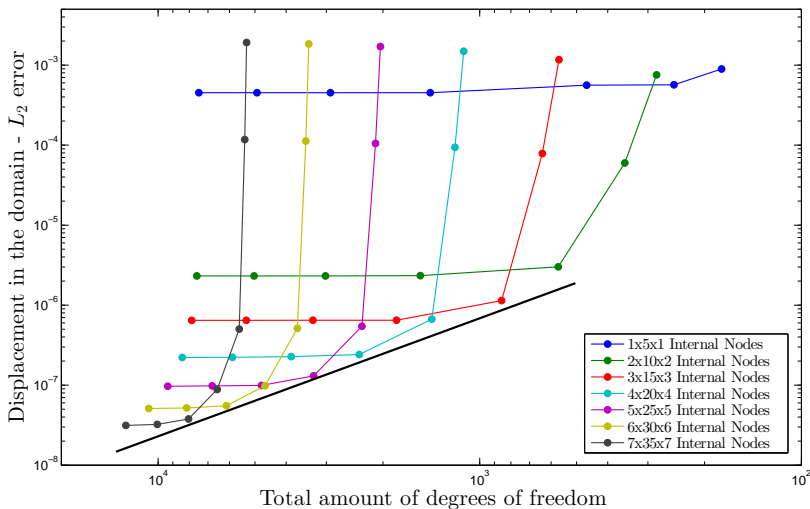
Figure 8.11: L_2 error of displacements in the domain

solving the resultant system of equations. Subsequently, in a postprocessing stage, the values of traction on the boundary for the distribution of colloca- tion nodes of each example is calculated. Finally, it is generated a equispaced distribution comprising $3 \times 15 \times 3$ internal points, where displacements and stress tensors are calculated.

Each graph presents error curves, where the amount of internal nodes remains constant and the number of elements on the boundary increases. Also, a line that represents, approximately, the trend of error for each calculated variable is drawn.

From the graphical analysis of results, it is seen that the theoretically expected behavior is maintained. Again the two asymptotes that reflect, respectively, the prevalence of errors associated with the approximation of the term $\hat{b}$ or with the boundary element approximation appear.

The evolution of the error is very stable and no conditioning problems
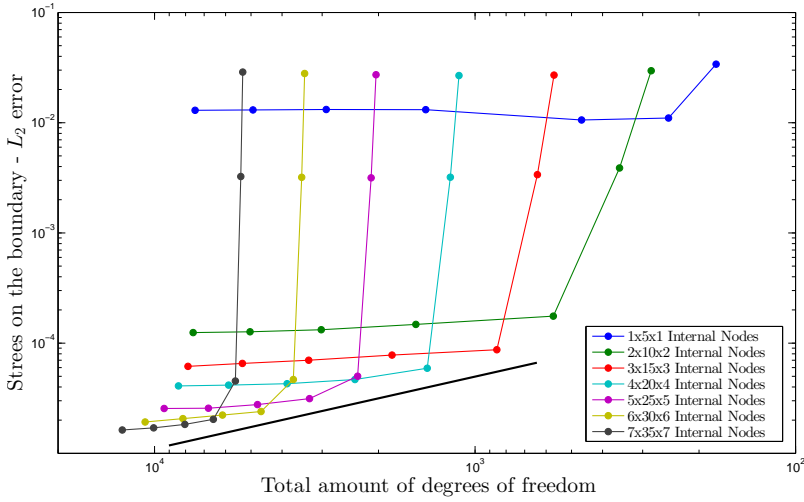
Figure 8.12: L_2 error of tractions on the boundary
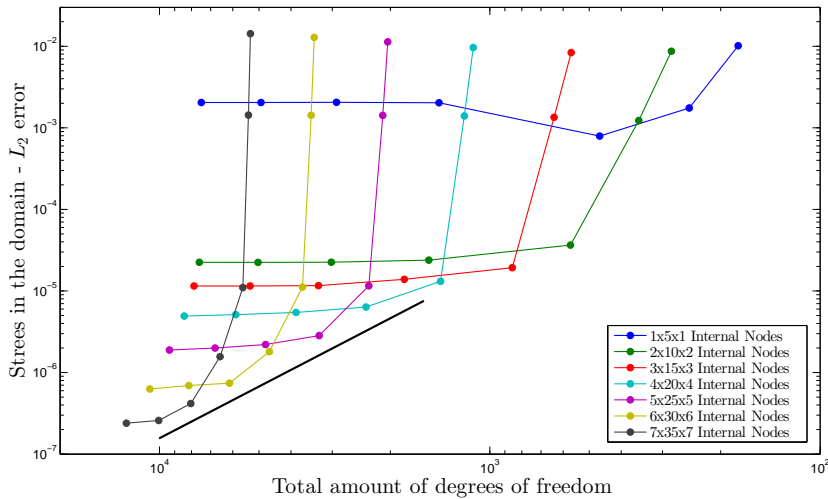
Figure 8.13: L_2 error of stresses in the domain
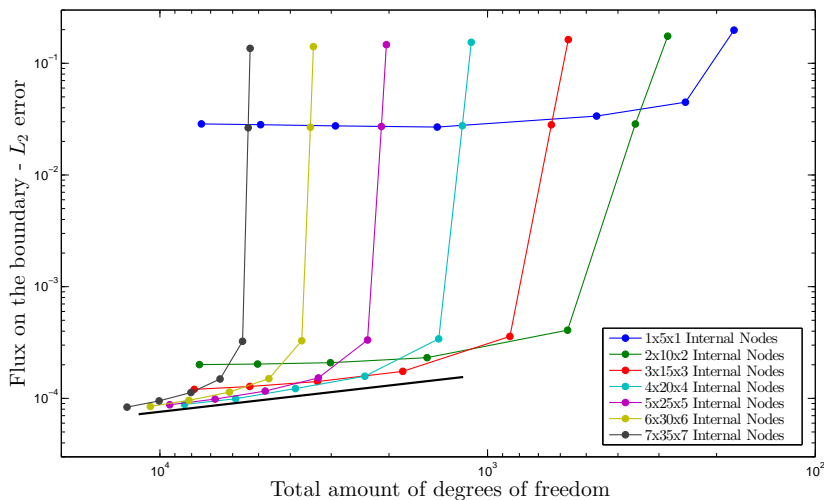
Figure 8.14: L_2 error of fluxes on the boundary

have been found in solving the resulting systems. The approximate rate of convergence for the internal points error displacement and stress is $1.6 - 1.8$. For fluxes and tractions on the boundary the rate of convergence of the error is 0.5. Note that, in this case, the error is concentrated on the edges of the faces of the prism and outside these small areas the error is much lower.

The final analysis performed for this example is, again, a comparative analysis of different types of approximations functions of the term $\hat{b}$. Specifically, the following approximation functions are compared

- ATPS 2D

- ATPS 3D

- Multiquadrics (MQ)

- Augmented Multiquadrics (AMQ)

- Linear
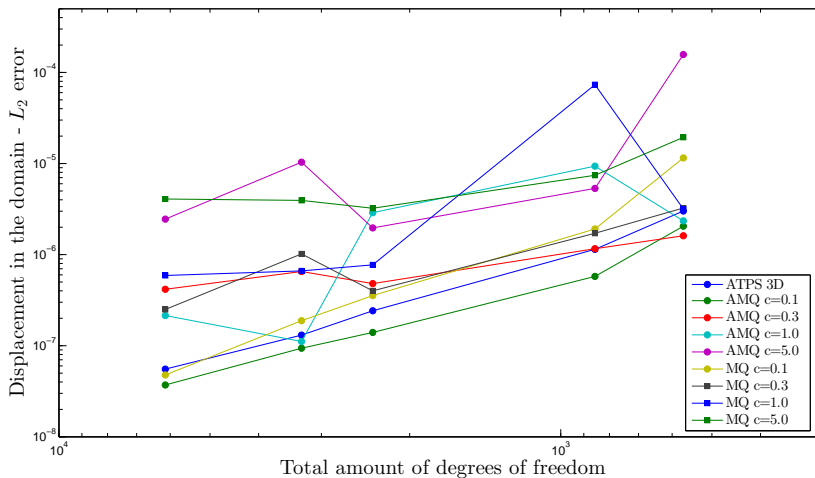
Figure 8.15: Error envelopes of displacement in the domain (a)

- Spline with several support values

- Second-order Wendland with several support values

As in the two-dimensional scalar example of the previous section, the relative behavior of the different approximations is maintained for every (internal displacement, flux ...) field. Due to this and, since 27 cases were compared, only comparative graphs of displacement in internal points are included.

In graphs 8.15, 8.16 and 8.17 is included, as reference, the error curve associated with ATPS 3D approximation. The analysis of these graphs allows to extract several ideas quite similar to those obtained in 2D problems (and in other 3D examples that have been executed using the code developed in this work).

- Approximation families of type $f(r) = c + r$ show the worst overall performance.

- Multiquadrics approximations, with and without polynomial terms,
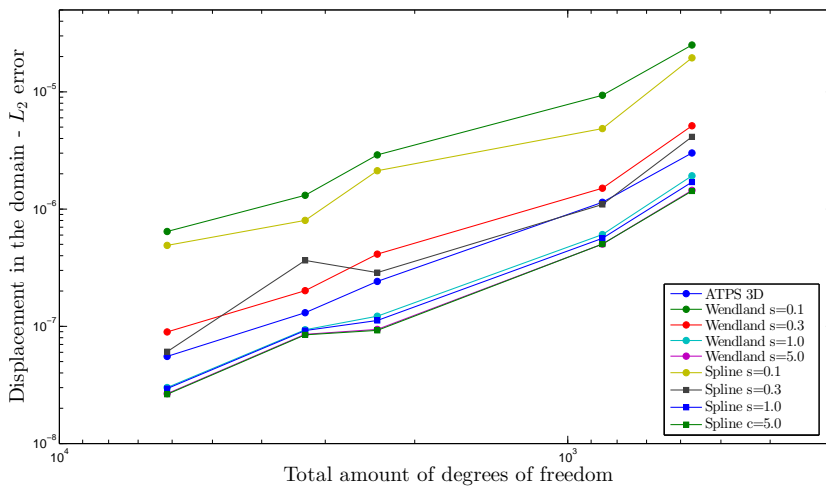
Figure 8.16: Error envelopes of displacement in the domain (b)
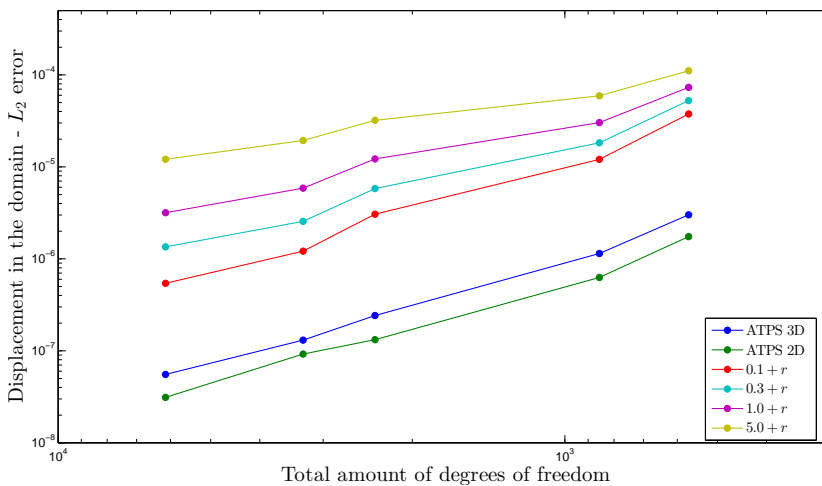
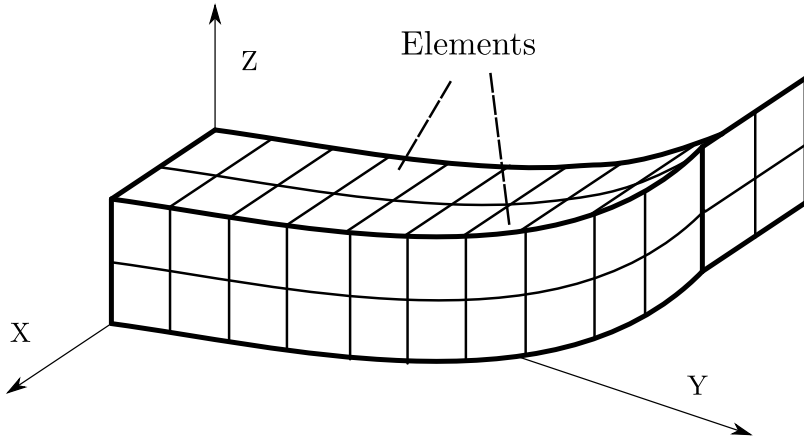Figure 8.17: Error envelopes of displacement in the domain (c)

Figure 8.18: Geometry and boundary discretization of a bent prism

are highly dependent on c parameter, both in terms of accuracy and stability of the resulting systems.

- In the case of approximations with support or dependent parameters, the accuracy of the results is strongly influenced by the value of these constants. Still, it should be noted that both types, Wendland and spline functions, present good results and allow local approximation implicitly. From the numerical tests performed a reference value of the support can be extracted. The value deduced is the order of magnitude of the typical dimensions of the problem to solve.

- The ATPS 2D approximation functions have shown again, the best balance of accuracy, stability and simplicity. In addition no further considerations (support or parameter values) are needed for its definition.

Geometry 2

The second example is a geometrical variation of the domain of the previous example. The variation of the properties and the displacements definitions are maintained. The objective of this example is to study the behavior of the algorithm in terms of accuracy and stability in a more complex discretization including curved elements.

The geometrical variation consists of curving the prism of the previous example along a quadratic path increasing the value of the z coordinate. The equations of the lower and upper curved edges in the $X = 0$ plane are given by $z = 0.40y^2$ and $z = 0.40y^2 + 0.20$, respectively.

The results are presented identically to the previous section by means of several graphs representing the L_2 error distribution versus the amount of degrees of freedom. In order to facilitate visual comparison of the evolution of the error, it has been superimposed for this purpose, the line that shows the rate of convergence of the error of previous example. Note, finally, that ATPS 3D is chosen again to approximate the fictitious body forces $\hat{b}$.

The analysis of the graphs shows that for internal points, the results, both in terms of accuracy and rate of convergence are virtually unchanged. In the case of fluxes and tractions on the boundary, the error has a significant increase although the trend continues in the case of the flux and is slightly deteriorated in the case of tractions. It is important to note, again, that most of this error is confined to areas close to the edges, and it is reduced markedly as we move away from these areas. This behavior is more pronounced the greater the angle of the edge is, which explains higher degradation of the error of these fields (flux and traction on the boundary).

The curves comparing the different approximation families present a identical behavior to the previous case and don't provide additional information. Therefore, they are not included for the sake of brevity.
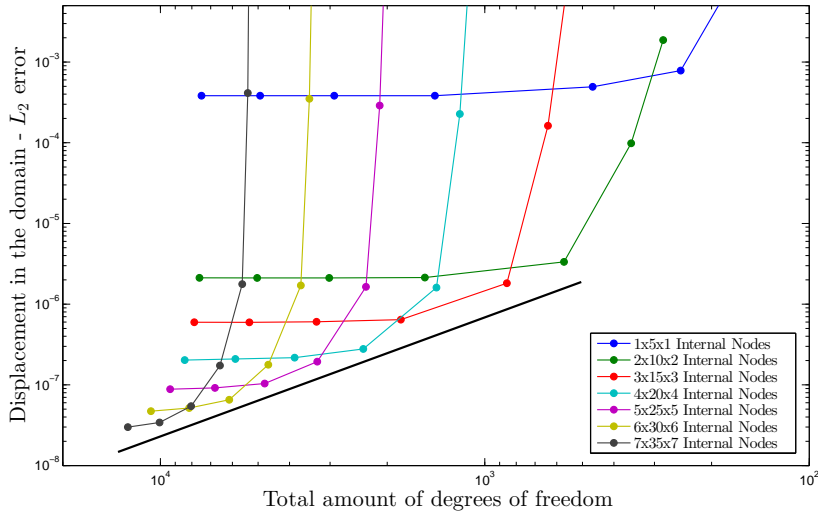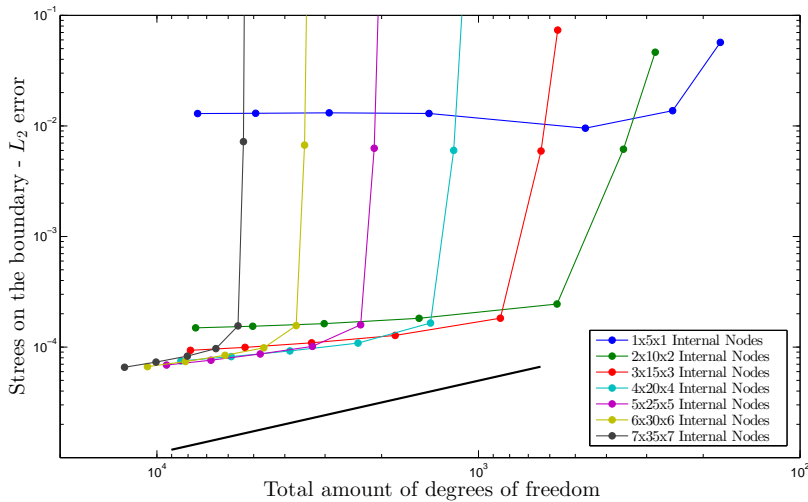
Figure 8.19: L_2 error of displacements in the domain

Figure 8.20: L_2 error of tractions on the boundary
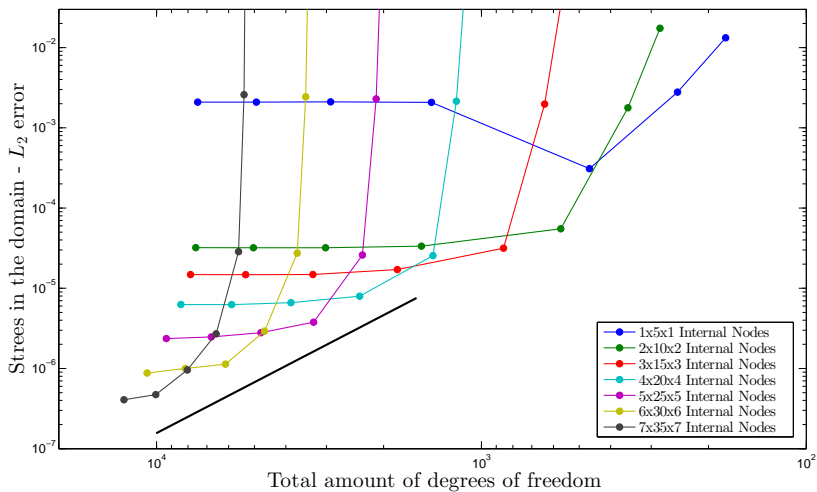
Figure 8.21: L_2 error of stresses in the domain
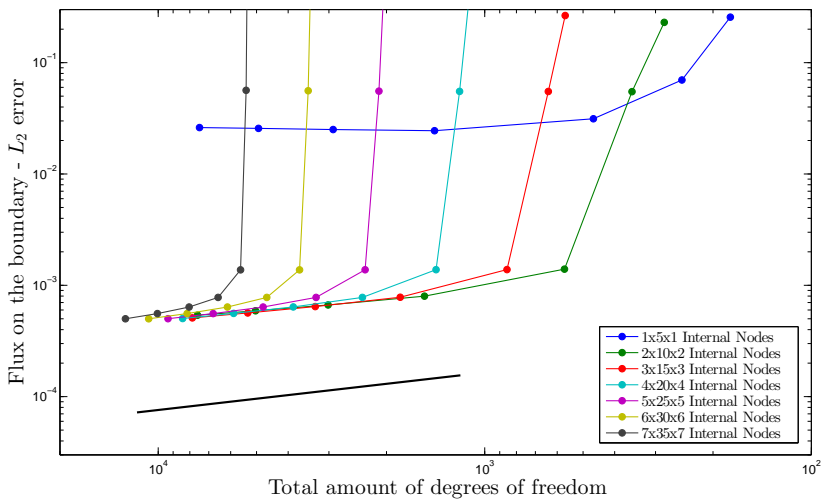

Figure 8.22: L_2 error of fluxes on the boundary
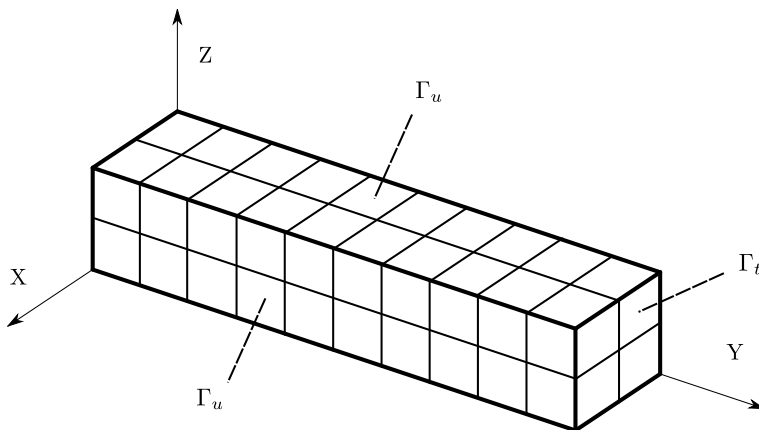
Figure 8.23: Boundary conditions and discretization of a prism

8.2.2   Mixed boundary conditions - 1

In this example a problem including traction boundary conditions is solved. The same displacement fields and variations of properties, as in the previous example, are used. So we have

$$u_x = \sin x; \qquad u_y = \cos y; \qquad u_z = e^z \tag{8.8}$$

$$\lambda = 100{,}000 - 50{,}000(x - y + z); \quad \mu = 80{,}000 + 80{,}000(x + y - z) \tag{8.9}$$

Geometry 1

The example to study has the same geometry as in the example presented in 8.2.1 (see figure 8.10). Therefore we are dealing with a prism of dimensions $X \times Y \times Z = 0.20 \times 1.00 \times 0.20$. In this case, as indicated, we have mixed boundary conditions. On the sides contained in planes $Y = 0.00$ and $Y = 1.00$ tractions boundary conditions are applied while on the remaining part of the boundary displacements are prescribed.
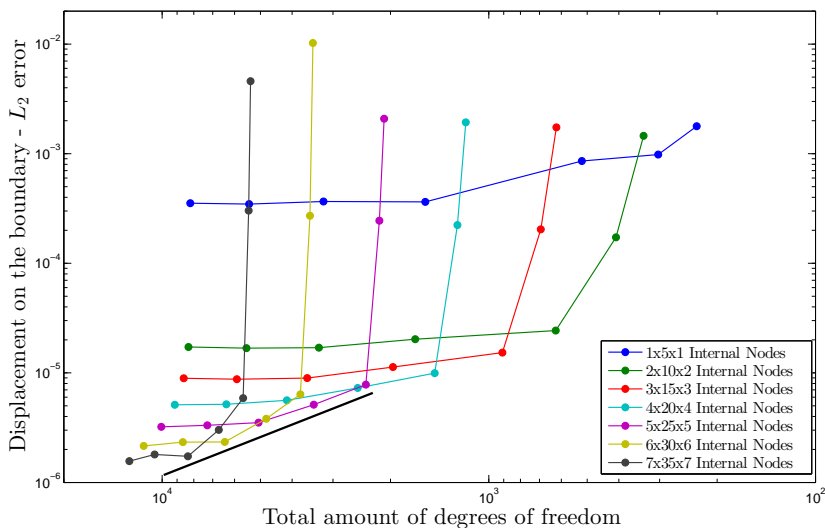
Figure 8.24: L_2 error of displacements on the boundary

The approximation function chosen remains the ATPS 3D to maintain consistency with previous analyzes. The results are presented by the same kind of graphs, but in this case, the errors associated with the displacements on the boundary are included. The rate of convergence of the error for each calculated field is also drawn approximately.

The analysis of the results shows that the expected behavior continues, although a degradation of the accuracy can be observed. The conditioning of the systems shown no problems.

Qualitatively the error of displacement on the boundary has a rate of convergence of $1.3 - 1.4$, while rate of convergence for the displacement and stress in the internal points is, approximately $1.5 - 1.6$. In the case of tractions and fluxes on the boundary the rate of convergence is maintained at 0.5. Note again, that the error is concentrated on the edges of the faces of the cuboid, and it is reduced markedly as we move away from these areas.
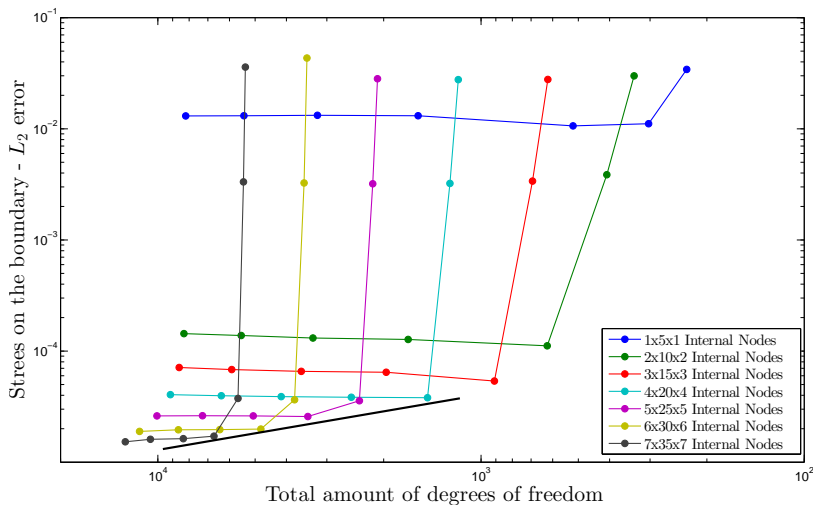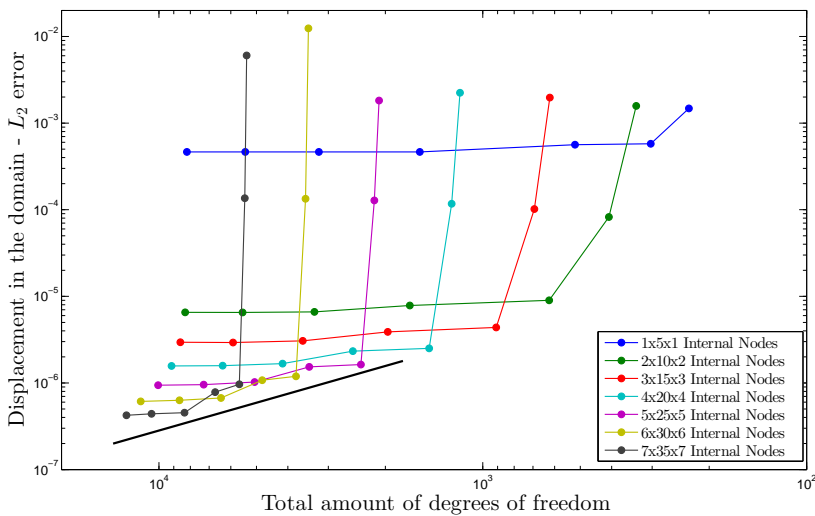
Figure 8.25: L_2 error of tractions on the boundary

Figure 8.26: L_2 error of displacements in the domain
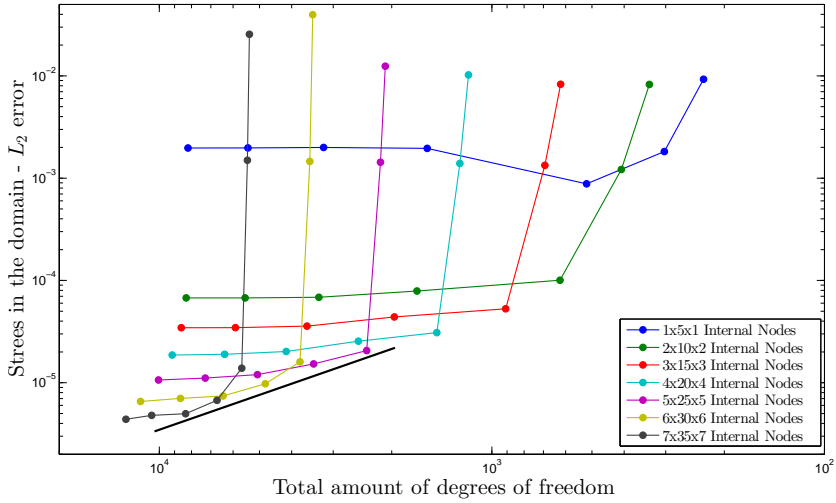
Figure 8.27: L_2 error of stresses in the domain

Figure 8.28: L_2 error of tractions on the boundary
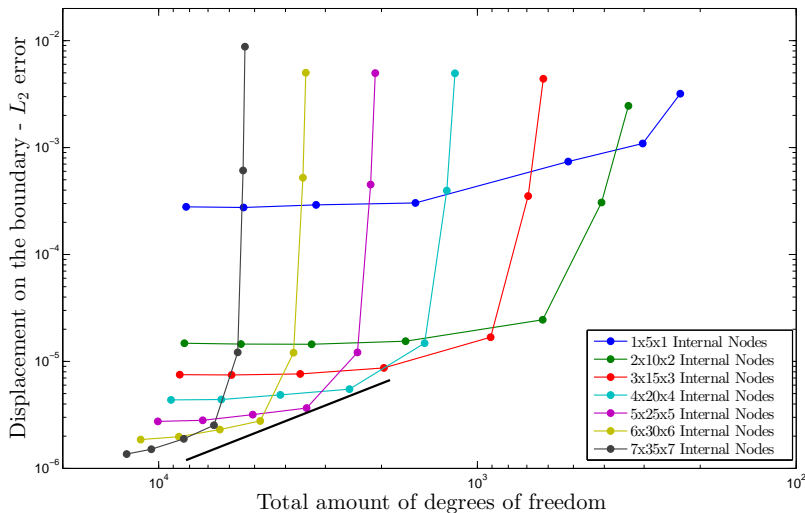
Figure 8.29: L_2 error of displacements on the boundary

For the sake of brevity, the comparative analysis of the different approximations functions is discussed in the next example.

Geometry 2

In the last example of this part the curved prism is discussed again, in this case with the same boundary conditions as in the previous example. In figures 8.29, 8.30, 8.31, 8.32 and 8.33 the errors of the five fields studied are represented. Lines representing the error rates of convergence of the previous example, have been superimposed to visually asses behavioral changes. Finally the approximation function chosen remains the ATPS 3D to maintain consistency with previous analyses.

Visually, it can be observed that changes in behavior are similar to the example with displacement boundary conditions. Thus, in the internal points,

Figure 8.30: L_2 error of tractions on the boundary
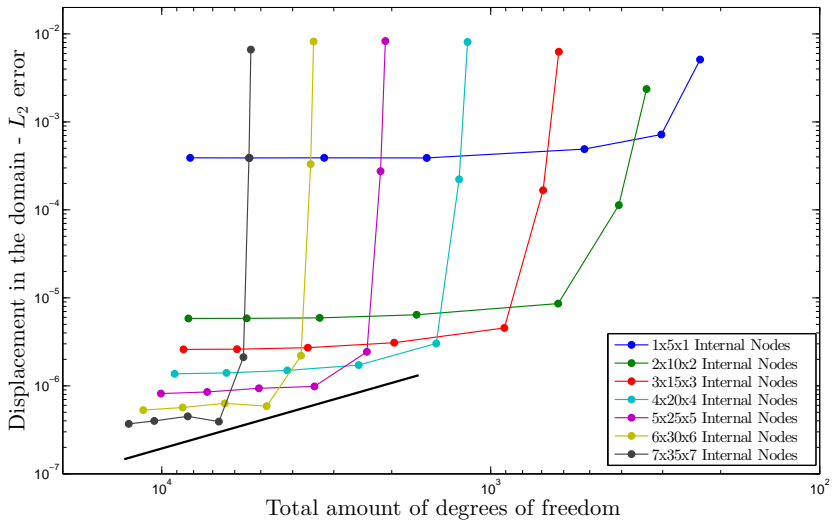
Figure 8.31: L_2 error of displacements in the domain
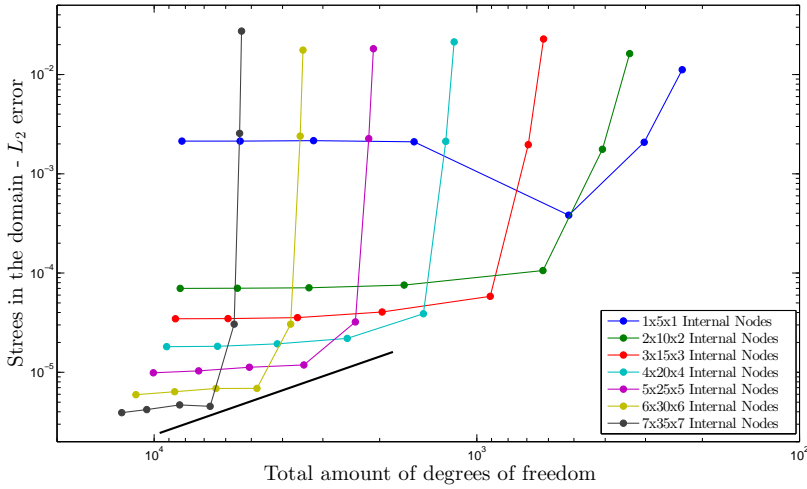
Figure 8.32: L_2 error of stresses in the domain
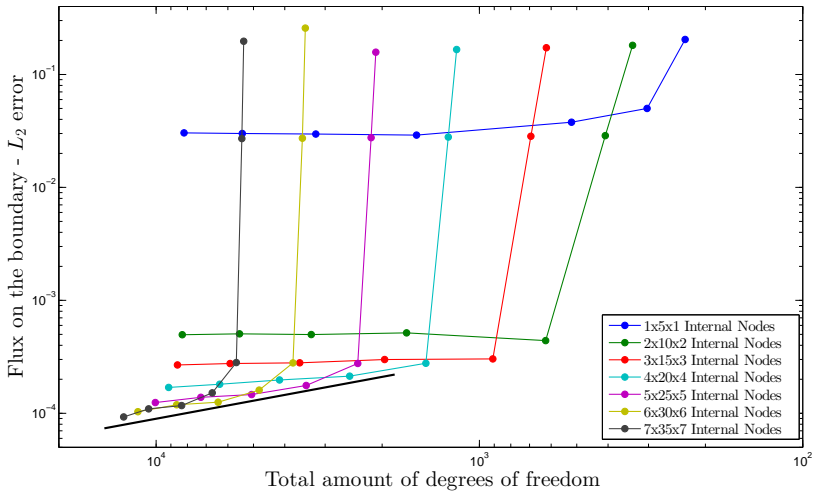

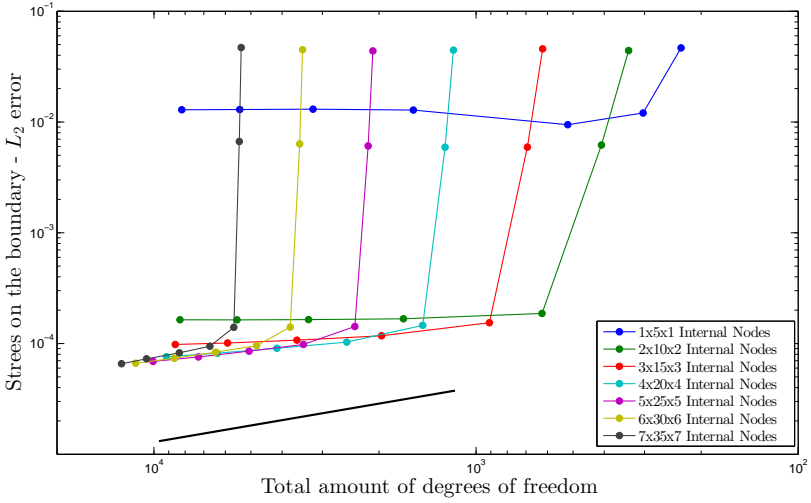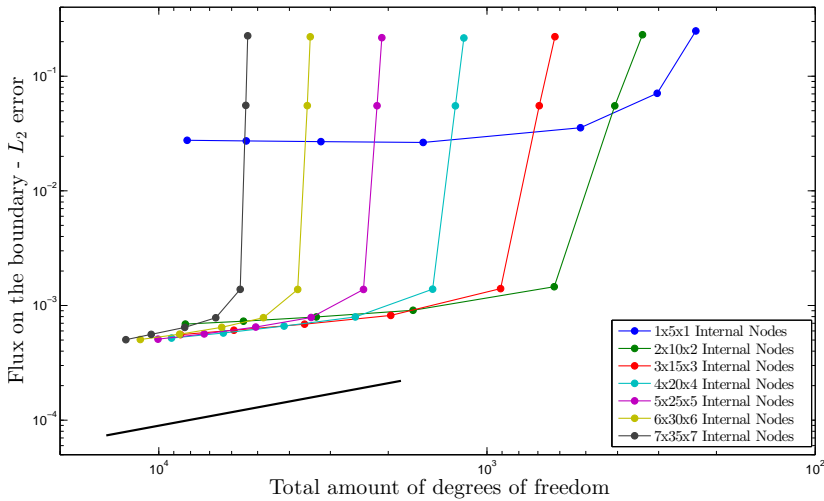Figure 8.33: L_2 error of tractions on the boundary
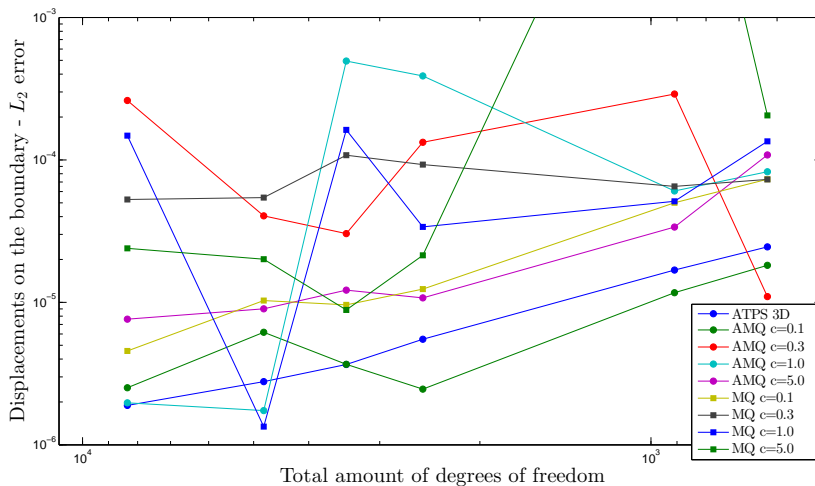
Figure 8.34: Error envelopes of displacement in the boundary (a)

the results, in terms of accuracy and rate of convergence are practically unchanged. In the case of fluxes and tractions on the boundary, the error has a significant increase although the trend continues in the case of the flux and is slightly deteriorated in the case of tractions.

The final analysis performed for this example is, again, a comparative analysis of different types of approximations functions of the term $\hat{b}$. The same functions as in the Dirichlet boundary conditions case are used. Only the comparison of one variable is presented because it is representative of the other fields. In this case, the selected field is the displacement on the boundary. In graphs 8.34, 8.35 and 8.36 is included, as reference, the error curve associated with ATPS 3D approximation.

The analysis of these graphs can draw several conclusions

- Approximation families of type $f(r) = c + r$ still present the worst overall performance, but its behavior is very stable.
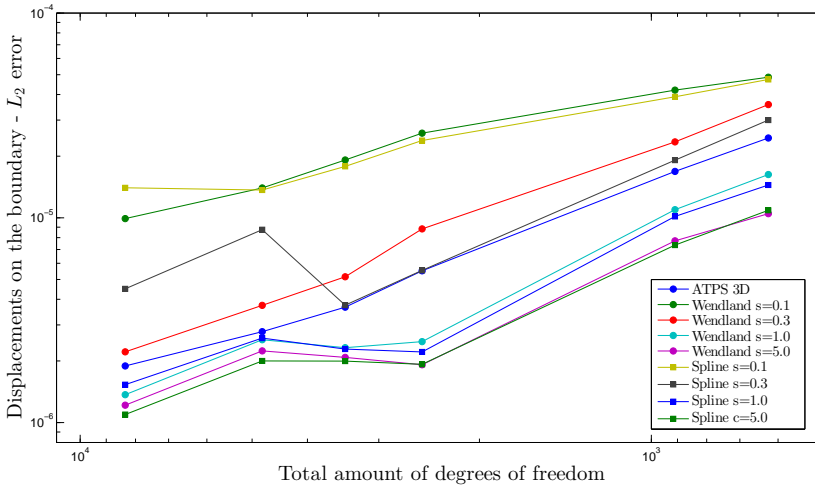
Figure 8.35: Error envelopes of displacement in the boundary (b)
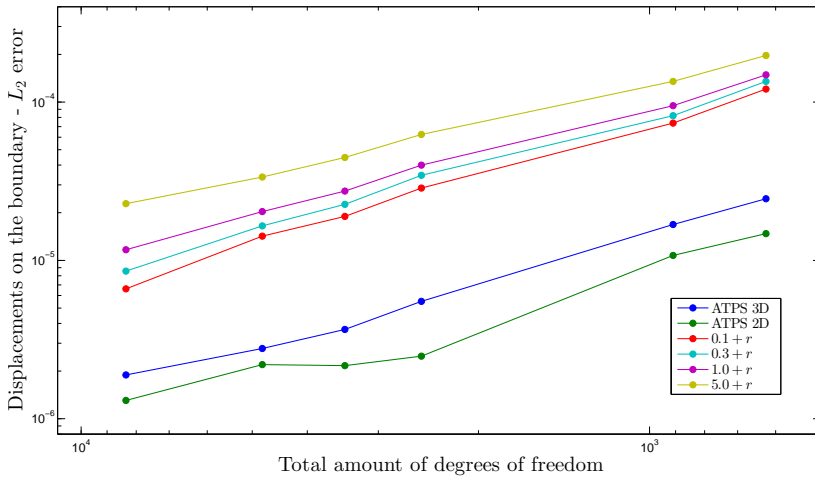
Figure 8.36: Error envelopes of displacement in the boundary (c)

- Multiquadrics approximations, with and without polynomial terms, have a totally erratic and hard of conditioning behavior.

- In the case of approximations with support or dependent parameters, the accuracy of the results is strongly influenced by the value of these constants. From the numerical tests performed a reference value of the support can be extracted . The value deduced is, again, the order of magnitude of the typical dimension.

- The ATPS 2D approximation functions have showed again, the best balance of accuracy, stability and simplicity. In addition no further considerations (support, parameter) are needed for its definition.

8.2.3   Mixed boundary conditions - 2

In this case a problem extracted from [42] is analyzed. A cuboid with dimensions of $X \times Y \times Z = 100mm \times 100mm \times 30mm$ is studied. Every surface except the one located on $Y = 100$ are placed on roller support so these surfaces can freely move in the tangential directions, but are constrained in the normal direction. The investigated cuboid is subjected to a uniform tensile loading $t = 1 \, dN/mm^2$ on the surface $Y = 100$ (see Figure 8.37). In this case the variation of the shear modulus μ is defined as

$$\mu = \mu_0 e^{\beta z} \quad \text{where} \quad \beta = \frac{\log(\mu_w/\mu_0)}{Z}$$
$$\mu_w = \ 8000 \, dN/mm^2 \qquad \mu_0 = \ 4000 \, dN/mm^2$$

(8.10)

while Poisson's ratio is constant with value 0.25. Body forces are not considered.

Horizontal displacement along a line running down the center of the top face (specifically $Z = 30$ mm, $X = 50$ mm and Y from 0 mm to 100 mm) is presented in figure 8.38. This displacement have been selected to compare the results obtained using the AEM-BEM algorithm with the results included in the work [42], where is presented a variant of BEM that combines
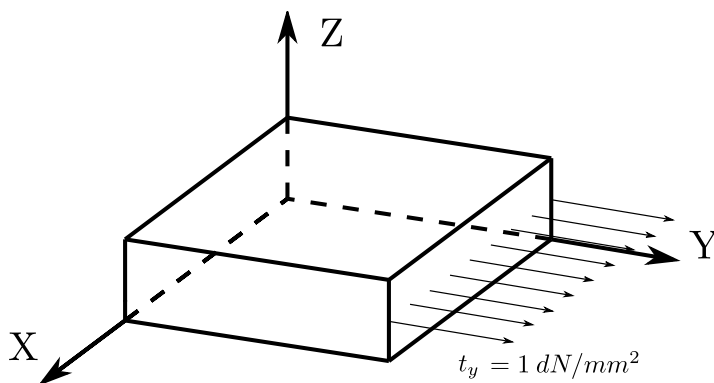
Figure 8.37: Cuboid under normal tensile loading on $Y = 100$

radial integrals with multidomains, in order to solve problems involving inhomogeneous domains.

Three algorithms are compared. A FEM model with 70000 linear tetrahedra, taken as a reference solution, has been performed with ABAQUS and is represented by the solid blue line. The example presented by Gao in [42] uses 8960 elements and 15048 internal nodes and is represented by the black line with the values marked by crosses. The results using the technique analyzed in this work (AEM-BEM) are shown by red square. A mesh of 70 quadratic elements in the boundary and a distribution of 300 internal nodes have been implemented to solve the example. The approximation function selected is ATPS 2D.

Analysis of the results shows the good agreement between the results using the Finite Element Method and the AEM-BEM methodology. The results presented by Gao shows a moderate difference, though clearly visible as moving in the direction Y. Taking into account the large number of degrees of freedom that are used in its calculation, it is possible that the definition of properties or the boundary conditions presented in the article [42], have some kind of definition error.
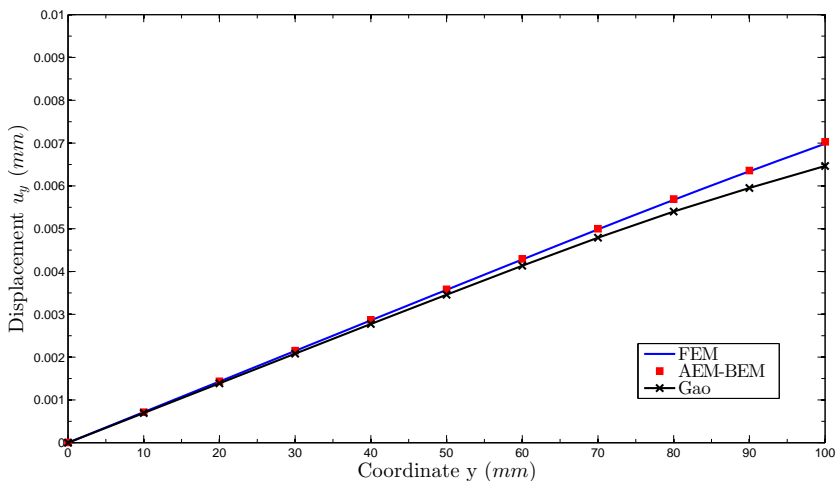
Figure 8.38: Comparison of the displacement at the center of the top face

8.3 Multidomains

8.3.1 Example with two subdomains

The first example studies a cube (see figure 8.39), consisting of two phases of different materials, dimensions of $X \times Y \times Z = 1.00 \times 1.00 \times 1.00$. The top half of the cube is formed of a inhomogeneous isotropic material with properties defined as

$$\lambda = 75{,}000 - 50{,}000z \quad \mu = 10{,}000 + 80{,}000z \qquad (8.11)$$

while the lower half, is formed of a homogeneous isotropic material with properties defined as

$$\lambda = 50{,}000 \quad \mu = 50{,}000 \qquad (8.12)$$
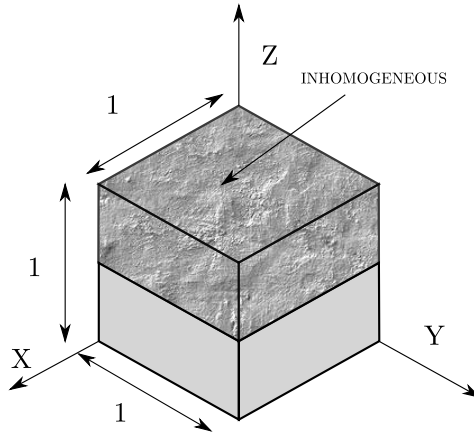
Figure 8.39: Bimaterial cube geometry

In this case, the displacement field and the properties of the domain are defined. With these data, it is simple to obtain the associated stresses and body forces that allow to define completely the problem. Also, fluxes (which, in this case, are variables without physical meaning), internal stresses or any desired magnitude can be obtained easily.

In order to evaluate the error accurately, a problem with known exact solutions is generated. In this case, we create a displacement field which together with defined material properties, satisfies the conditions of perfect contact at the interface.

The displacement field is defined as

$$u_x = ze^x \sin x; \qquad u_y = ze^x \cos y; \qquad u_z = z \qquad (8.13)$$

The boundary conditions on the external boundaries are of mixed type, with displacement boundary conditions on the upper and lower faces and tractions on the sides. The discretization of the boundary consists of four elements per side and four elements at the interface. In the inhomogeneous
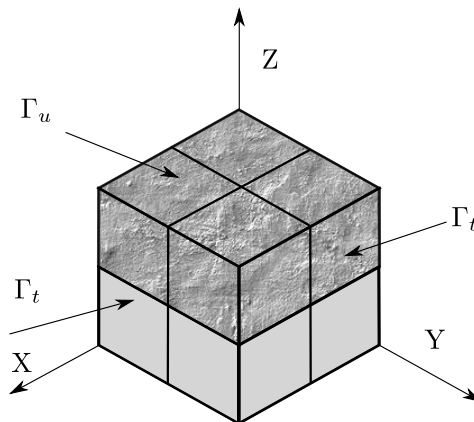
Figure 8.40: Boundary conditions and discretization

subdomain a distribution of $6 \times 6 \times 3$ internal nodes is used. Figure 8.40 shows the discretization scheme of the boundary. In this case, the problem is solved for a unique combination of boundary elements and internal nodes, but multiple functions approximation. After the resolution of the system the internal displacements and stresses on both sides of the cube are calculated.

Table 8.1 includes the L_2 error of displacements and tractions on the boundary along with displacements and stresses in the domain[3]. In the calculation of these errors are included both the values associated with the homogeneous part as the inhomogeneous one. The analysis of the results shows that, except for high values of the parameter c associated with AMQ (augmented multiquadrics) approximation functions, where there have been problems of conditioning, the error value is virtually identical for all approximation functions used. This is because, for the combination boundary element-internal nodes used in this example, the main factor of error comes from the calculation of the homogeneous part.

---

[3]In this case, the fluxes have not been included in the results due to its lack of physical meaning.

Type of Approximation	u_c L_2 error	t_c L_2 error	u_i L_2 error	σ_i L_2 error
ATPS2D	3.56E-03	5.04E-03	1.38E-03	2.26E-03
ATPS3D	3.56E-03	5.05E-03	1.39E-03	2.26E-03
Spline s=0.2	3.72E-03	5.24E-03	1.49E-03	2.37E-03
Spline s=1.0	3.44E-03	4.98E-03	1.33E-03	2.19E-03
Wendland s=0.2	3.73E-03	5.26E-03	1.48E-03	2.38E-03
Wendland s=1.0	3.46E-03	4.99E-03	1.35E-03	2.21E-03
AMQ c=0.1	3.52E-03	5.02E-03	1.37E-03	2.24E-03
AMQ c=1.0	1.23E-02	1.84E-02	6.34E-03	1.65E-02
AMQ c=10.0	8.70E-03	1.12E-02	2.79E-03	5.55E-03

Table 8.1: L_2 error for different approximations
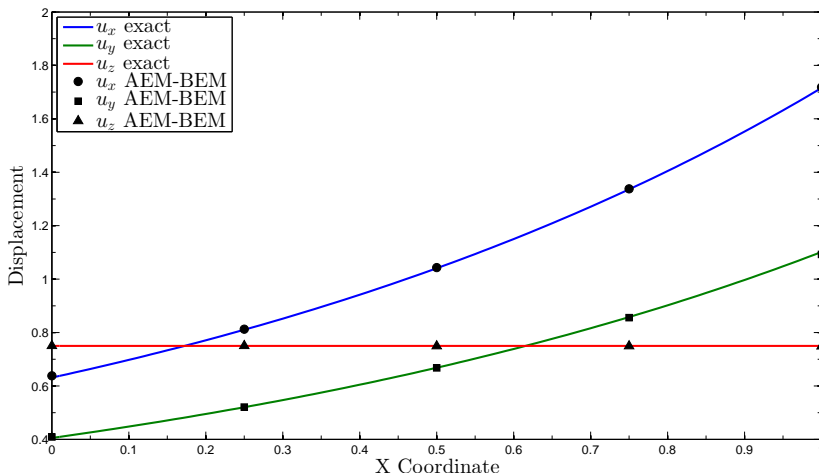
Figure 8.41: Displacements on the boundary $Y = 1.0$ $Z = 0.75$

In figures 8.41, 8.42, 8.43 and 8.44 the numerical results obtained using the ATPS 2D approximation function are represented and compared with the analytical solution for the different fields studied. The curves represent the variation of these fields along lines going through different parts of the domain, as indicated in each graph. The level of agreement is very high, as expected in view of the L_2 error presented in the previous table.

8.3.2   Example with three subdomains

The last example presented in this thesis is a three-dimensional version of an example included in [100], where an extension of the boundary element method, able to solve problems involving multidomains and inhomogeneous materials is introduced.

The multidomain studied in this example consists of three subdomains. The first one (referred as part 1) is a cuboid of dimensions $X \times Y \times Z = 20 \times 10 \times 10$
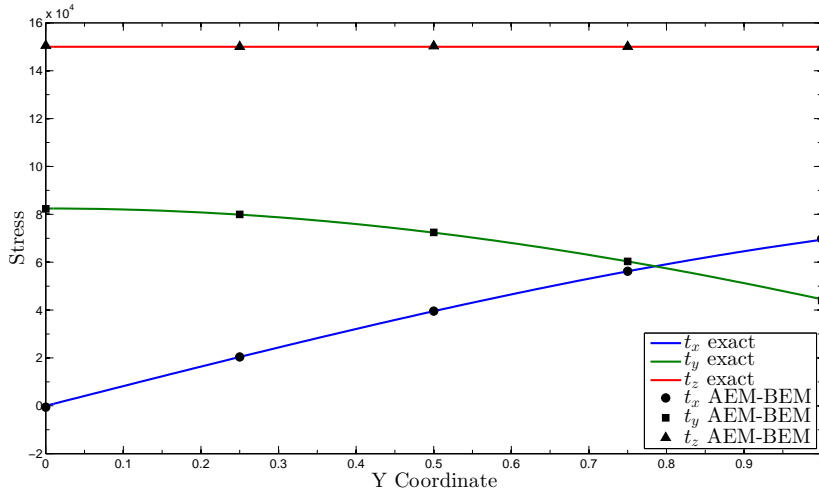
Figure 8.42: Stresses in the interface $X = 0.5$ $Z = 0.5$
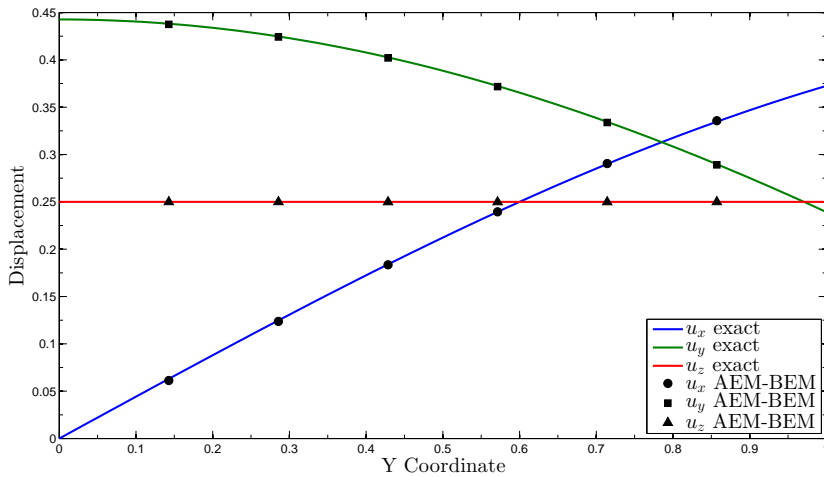
Figure 8.43: Displacements inside the cube $X = 0.5$ $Z = 0.25$
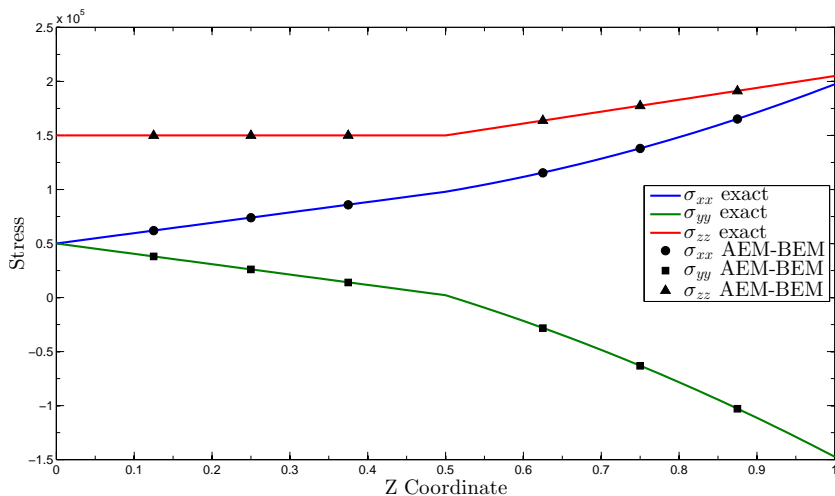
Figure 8.44: Principal stresses inside the cube $X = 0.5$ $Y = 0.5$

with a hole of radius 2, which runs the entire length of the cuboid. As boundary conditions it is assummed that the botom face is simply supported, the top face is subjected to a normal constant load and the remaining faces are free surfaces.

In figure 8.45 an outline with the indicated dimensions is attached. This part is designed as isotropic and homogeneous and its properties for all cases analyzed are $E_3 = 1.0$ and $\nu = 0.25$.

The second subdomain (referred as part 2) is a hollow cylinder of inside radius 1 and outer radius 2, which fits into the hole of the cuboid and, finally, a cylinder (referred as part 3) of radius 1 which fits into the hollow of part 2. These two pieces are shown in Figure 8.46.

Five different cases, with different properties for parts 2 and 3 are studied. In all cases, the Poisson ratio value will be constant and equal to $\nu = 0.25$ and perfect contact conditions between the three parts is considered. Part 3 is considered isotropic and homogeneous and its properties are $E_1 = 1.0$ in
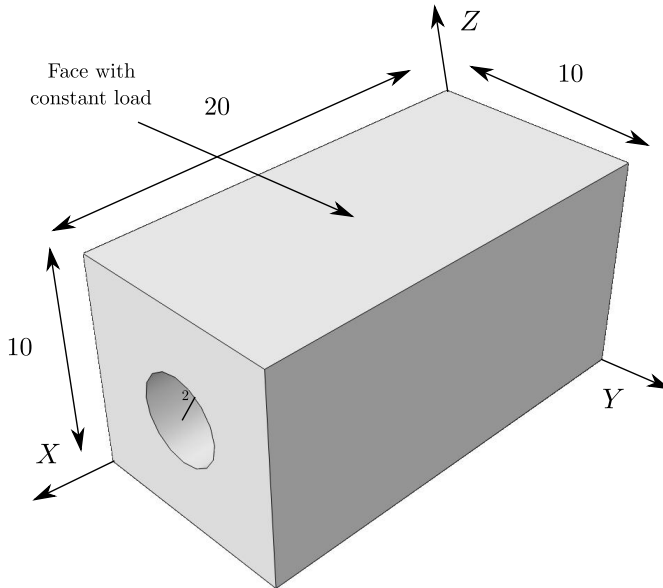
Figure 8.45: Geometry and dimensions of the prism
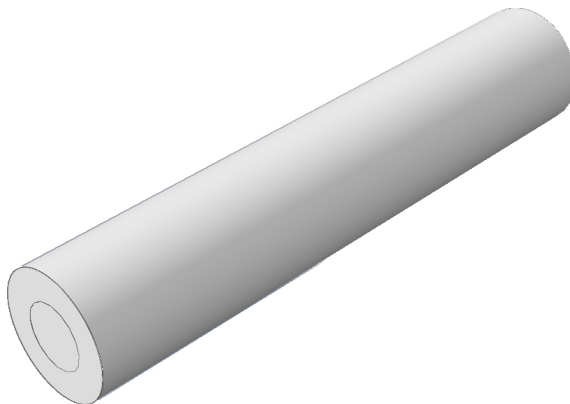
Figure 8.46: Parts 2 and 3 outline
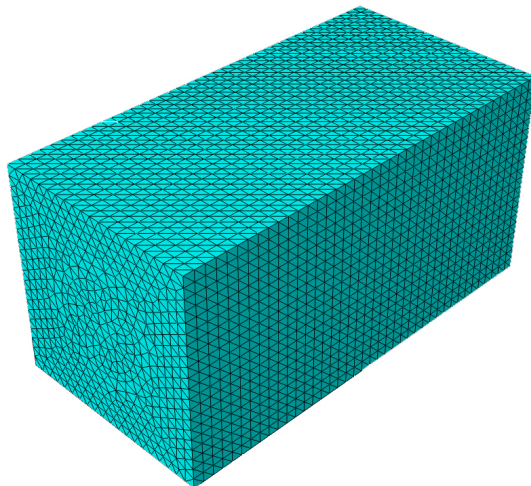
Figure 8.47: Discretization using finite elements

case 1 and $E_1 = 0.2$ in the remaining. The properties of the part 2 varies in each case as follows

- Case 1: Part 2 isotropic homogeneous $E_2 = 1.0$

- Case 2: Part 2 isotropic homogeneous $E_2 = 1.0$

- Case 3: Part 2 isotropic inhomogeneous with quadratic variation of Young Module $E_2 = -2.2 + 3.2r - 0.8r^2$

- Case 4: Part 2 isotropic inhomogeneous with linear variation of Young Module $E_2 = -0.6 + 0.8r$

- Case 5: Part 2 isotropic homogeneous $E_2 = 0.2$

Two algorithms are compared. Firstly, a FEM model with 97000 elements (quadratic tetrahedra) has been performed with ABAQUS (in figure 8.47 the discretization scheme is shown). Secondly the AEM-BEM methodology studied in this thesis. In this case a mesh of 90 quadratic elements
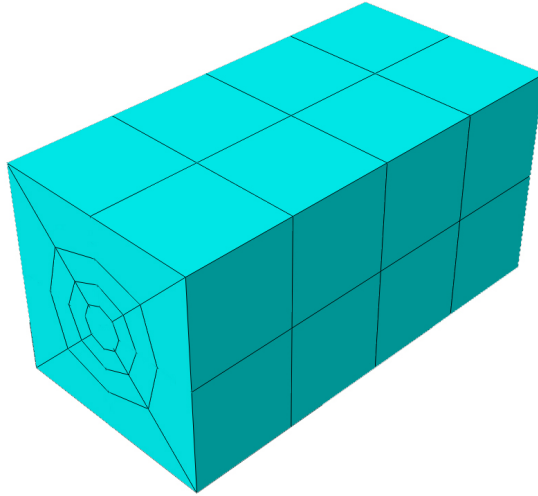
Figure 8.48: Discretization using boundary elements

(including interfaces) and a distribution of $10 \times 6 \times 3$ internal nodes[4] (in the inhomogeneous part) is used. Figure 8.48 the discretization scheme using boundary elements is shown[5].

In cases where all pieces are homogeneous, standard BEM (with multidomain) is used to solve the problem, and their inclusion in this example allows, firstly, to check the implementation of this scheme in the software, and, on the other hand, to analyze variations in the results depending on the different properties.

In view of the results of previous sections, ATPS 2D have been chosen as approximation function in this example. Although not shown in the results, this example has served to verify that the approximation functions work

---

[4]This distribution must be understood as longitudinal direction $\times$ circumferential direction $\times$ radial direction.

[5]The elements in the front (and rear) face are quadratic and their real boundaries are not straight, as figure 8.48 suggests.

better when they are scaled with the order of magnitude of the typical dimensions of the problem to be treated. This means that the function used in this example is

$$f(r) = \left(\frac{r}{s}\right)^2 \ln\left(\frac{r}{s}\right) \quad \text{where} \quad s = 10 \tag{8.14}$$

Finally, it should be pointed that in order to avoid errors in the curved boundaries, all collocation nodes associated with inhomogeneous boundaries have been moved inside the elements. The reason of this is to avoid the errors associated with the discontinuities of the approximation functions and the normal vectors between elements (see Chapter 6 for more references).

Two graphs of results are presented: the displacement u_z and the normalized Von Misses stress, for a line that cuts across the domain from $(X = 10, Z = 5, Y = 0)$ to $(X = 10, Z = 5, Y = 10)$.

In figure 8.49 displacements of the five cases are presented in a single graph, where it can be observed that despite the coarse mesh used, the concordance of the results is very high.

The graphs associated with stresses have been divided into two, to clarify the analysis. The results of the homogeneous cases are included in graph 8.50 while in graph 8.51 are included the results corresponding to the inhomogeneous cases (case 1 has remained as a reference). Although the biggest differences are in the last graph, the correlation is quite good and the maximum values are well represented. Finally it must be noted that the values obtained using finite elements have some uncertainty because, for example, the stress distributions are not even symmetrical.
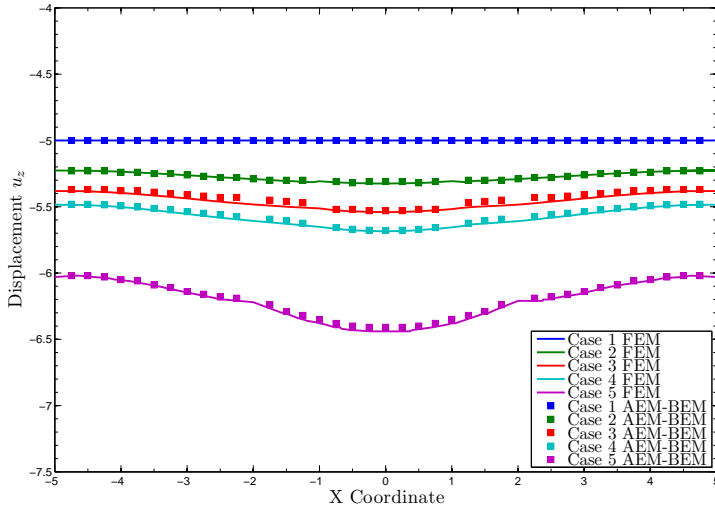
Figure 8.49: Displacement u_z in all cases
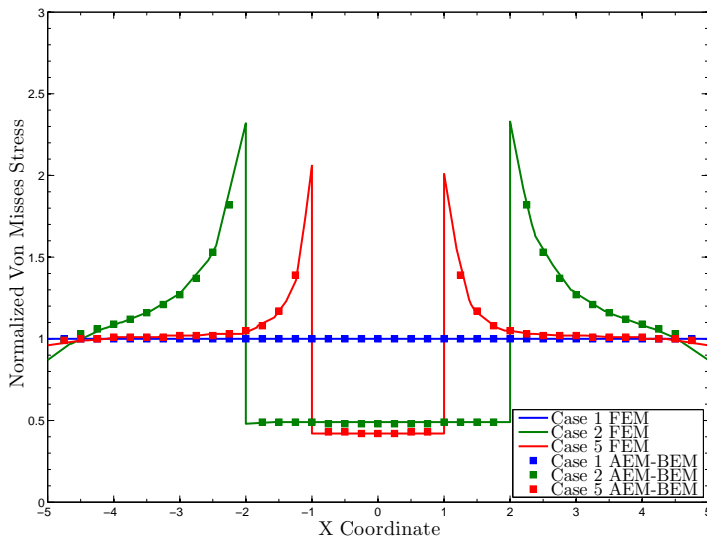
Figure 8.50: Normalized Von Misses stress in homogeneous cases
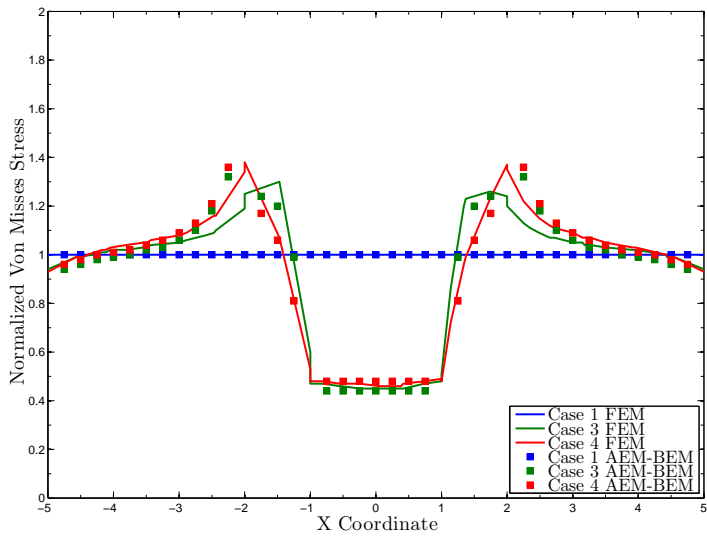
Figure 8.51: Normalized Von Misses stress in inhomogeneous cases

The whole sense of the book might be summed up the following words: what can be said at all can be said clearly, and what we cannot talk about we must pass over in silence.

Ludwig Wittgenstein

9

Conclusions

This dissertation has studied mainly, two sets of objectives:

- Study the behavior of the AEM-BEM methodology, and generalize it including an analytical formulation using general analog operators, introducing a integral formulation (**q-BIE**) that allows input directly boundary conditions derived from the main variables, characterizing this formulation to three dimensional elastic problems, performing a comparative study of the different families of approximation functions and coupling it with the standard BEM methodology.

- Build the core of an object-oriented program based on the latest iteration of FORTRAN (2003/2010) for, besides the implementation of this algorithm, easily scale and integrate in a natural way different BEM techniques.

9.1 AEM-BEM Methodology Conclusions

Regarding the first set of objectives

- The AEM-BEM methodology is able to accurately resolve problems involving inhomogeneous materials such as FGM, even using a relatively small number of internal domain nodes and boundary elements.

The algorithm is convergent and stable for most of the approximation functions.

- Generally, it has been proved that the domain approximation functions that exhibit better behavior in terms of stability and precision are the ATPS type both in 2D and 3D problems. Additionally, they have the advantage of the absence of parameters that must be adjusted.

- The behavior of the domain approximation functions is significantly improved if they are scaled, at least, using the order of magnitude of the problem typical dimension.

- The Wendland and Spline approximation functions also show good behavior and allow us to introduce local approximation implicitly. As drawback the accuracy of the results strongly depends on the value of the support and there are no general rules for determining this value. A support value in the order of magnitude of the problem is a good start reference. Smaller values imply greater locality of the approximation.

- Multiquadrics functions and its derivatives have a very unstable behavior and they have been discarded for later studies.

- A proportion of 50% (internal nodes DOFs – total DOFs) has proved to have a good behavior as suggested distribution.

- In the case of he boundaries that can not be smoothly approximated using quadratic elements, if Neumann boundary conditions are used, the placement of the collocation nodes inside the element significantly improves the accuracy avoiding continuity errors.

- The coupling of the standard BEM with AEM-BEM methodology maintains high levels of accuracy and reduces the number of degrees of freedom of multiphase problems by limiting the AEM-BEM algorithm area of application to inhomogeneous areas.

9.2 Object-oriented implementation Conclusions

- The proposal for the design of software architecture reproduces, naturally, the geometries and problems manipulation used in the theoretical analysis.

- The use of encapsulation techniques ensures that subsequent additions do not affect the previous implementations, allowing the growth of the program ensuring its integrity.

- Integrate new problems requires only a few simple steps. The addition of a class with the parameters of the problem, the definition of the differential operator including its variables and derivatives, an associated transfer matrix and an object that includes the integral derived from the singular part of the fundamental solution kernel.

- Fortran 2003/2010 backwards compatibility allows easily the use of the large base of existing algorithms written in previous FORTRAN iterations as the basis of the objects developed for the object oriented style software.

9.3 Future Works

Future developments will involve the development of both lines of work, although some of them are not confined strictly to the AEM-BEM implementation. Briefly.

- Implementing a fast solver like GMRES [101] is a priority, since using large numbers of degrees of freedom, the main part of the computation time is consumed on the matrix inversion.

- Integrate AEM-BEM methodology for 2D problems within object-oriented software porting it from Matlab.

- Implement new static problems such as poroelastic, thermoelastic ...

- Develop automatic geometry transfer schemes to connect to standard mesh definition formats.

- Implement dynamic problems using step-by-step schemes.

- Implement complex numbers based schemes to address problem types like Helmholtz type, elastodynamics in frequency ...

- Implement schemes for solving nonlinear problems.

- Implement schemes to solve general fracture problems.

- Study the applicability of FAST BEM schemes to the AEM-BEM methodology, in particular, Fast Multipole Method [89] and Adaptive Cross Approximation [99], [12].

- Implement Galerkin type methodology.

- Study the parallelization of the code to reduce calculation times.

- Implement NURBS type approximation schemes [114] to avoid the problems of discontinuity of the shape functions.

9

Conclusiones

Este trabajo ha estudiado, en líneas generales, dos grandes grupos de objetivos:

- Estudiar el comportamiento de la metodología AEM-BEM, generalizando su formulación analítica incluyendo funciones análogas generales, introduciendo una formulación integral que permita introducir directamente condiciones de contorno en derivadas de las variables principales, desarrollando esta formulación para problemas elásticos tridimensionales, realizando un estudio comparativo de las diferentes familias de funciones de aproximación y acoplando este algoritmo con la metodología BEM estandar.

- Construir el núcleo de un software orientado a objetos basado en la ultima iteración de FORTRAN (2003/2010) para, además de la implementación de este algoritmo, construir una base escalable que permita integrar, de forma orgánica, diferentes esquemas de cálculo tipo BEM.

9.1 Conclusiones de la Metodología AEM-BEM

Respecto del primer grupo de objetivos

- La metodología AEM-BEM es capaz de resolver de forma precisa problemas que involucran materiales inhomogéneos como los FGM, incluso usando un número relativamente pequeño de nodos de aproximación y elementos de contorno. El algoritmo es convergente y estable para la mayor parte de las funciones de aproximación.

- En líneas generales, las funciones de aproximación que presentan un mejor comportamiento en términos de estabilidad y precisión son las funciones tipo ATPS tanto en problemas 2D como 3D. Adicionalmente, presentan como ventaja la ausencia de parámetros que deban ser ajustados.

- El comportamiento de las funciones de aproximación mejora notablemente si están escaladas, como mínimo, por el orden de magnitud de la dimensión típica del problema a resolver.

- Las funciones tipo Wendland o Spline también presentan un buen comportamiento y nos permiten introducir una aproximación de tipo local de manera implícita. Como contra la precisión de los resultados depende del valor del soporte y, no existen reglas generales para determinar este valor. Como regla general un soporte del orden de magnitud del problema es un buen valor de referencia. Valores menores implican mayor localidad de la aproximación.

- Las funciones multicuadráticas y derivadas presentan un comportamiento muy inestable y han sido descartadas para estudios posteriores.

- Una proporción del 50 % (Nodos internos – Grados de libertad totales) resulta adecuada a falta de mejores consideraciones.

- En los contornos no aproximables mediante elementos cuadráticos, si se utilizan condiciones en tensión, el retranqueo de todos los nodos de colocación mejora notablemente los resultados al evitar los errores de continuidad.

- El acoplamiento del BEM estandar con el AEM-BEM mantiene altos niveles de precisión y, permite reducir el número de grados de libertad

de problemas multifase al reducir la aplicación del algoritmo AEM-BEM a las zonas inhomogéneas.

9.2 Conclusiones de la implementación orientada a objetos

- La arquitectura propuesta para el diseño del software reproduce, de forma natural, la forma de manipular geometrías y problemas que se utiliza en los análisis teóricos

- La integración de nuevos problemas sólo requeriría el añadido de una clase con los parámetros del problema, la definición del operador diferencial, sus variables y derivadas, una matriz de transferencia asociada y un objeto que incluya las integrales singulares.

- El uso de las técnicas de encapsulamiento asegura que los posteriores añadidos no afecten a las implementaciones previas permitiendo el crecimiento del programa asegurando la integridad del mismo.

- La retrocompatibilidad del FORTRAN 2003/2010 permite el uso de la gran base de algoritmos existentes en iteraciones anteriores del lenguaje, como base de los objetos utilizados.

9.3 Trabajos a desarrollar

Los desarrollos futuros implican el desarrollo de ambas líneas de trabajo, aunque ciertos desarrollos no se circunscriben de forma estricta a la impementación del algoritmo AEM-BEM. De forma resumida.

- Dado que a grandes números de grados de libertad, la mayor parte del tiempo de cálculo se centra en la inversión de la matriz, la implementación de un solver tipo GMRES [101] es una prioridad.

- Integrar el esquema AEM-BEM para problemas 2D dentro del software orientado a objetos desde Matlab.

- Implementar nuevos problemas tipo estático como el elastotérmico, poroelástico...

- Automatizar los esquemas de transferencia de geometrías para conectar con formatos de definición de malla estandar.

- Implementar problemas dinámicos mediante esquemas paso a paso.

- Introducir esquemas basados en números complejos para abordar problemas tipo Helmholtz, elastodinámicos en el campo de la frecuencia...

- Introducir esquemas para la resolución de probemas no lineales.

- Introducir esquemas para la resolución de probemas de fractura generales.

- Estudiar la aplicabilidad de los esquemas tipo FAST BEM a la metodología AEM-BEM, en particular, los métodos multipolo [89] y Adaptive Cross Approximation [99], [12].

- Estudiar la paralelización del código para reducir los tiempos de cálculo.

- Implementar esquemas de aproximación tipo NURBS [114] para evitar los problemas de discontinuidad de las funciones de forma.

A

Kernels and limits

A.1 Limiting process

The assumption of smooth boundary around $\mathbf{z}$ is used in what follows.

In the case of the 3D Laplace operator, taking into account that Γ_ε is defined in our case as a sphere surface, using polar coordinates we can write
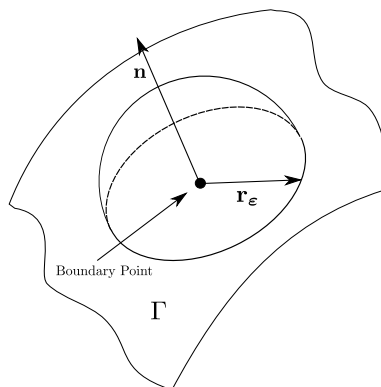
Figure A.1: Hemisphere around a boundary point at $\mathbf{z}$

$$\int_\Gamma w^* \left(\mathbf{x}; \mathbf{z} \right) \frac{\partial \phi}{\partial \mathbf{n}} \left(\mathbf{x} \right) \mathrm{d}\Gamma = \int_\Gamma \frac{1}{4\pi r} \frac{\partial \phi}{\partial \mathbf{n}} \left(\mathbf{x} \right) \mathrm{d}\Gamma =$$

$$\lim_{\varepsilon \to 0} \left(\int_{\Gamma - \Gamma_\varepsilon} \frac{1}{4\pi r} \frac{\partial \phi}{\partial \mathbf{n}} \left(\mathbf{x} \right) \mathrm{d}\Gamma \right) + \lim_{\varepsilon \to 0} \left(\int_{\Gamma_\varepsilon} \frac{1}{4\pi r} \frac{\partial \phi}{\partial \mathbf{n}} \left(\mathbf{x} \right) \mathrm{d}\Gamma \right) = \qquad \boxed{\text{A.1}}$$

$$\fint_\Gamma \frac{1}{4\pi r} \frac{\partial \phi}{\partial \mathbf{n}} \left(\mathbf{x} \right) \mathrm{d}\Gamma + \lim_{\varepsilon \to 0} \left(\int_{\theta,\varphi} \frac{1}{4\pi\varepsilon} \frac{\partial \phi}{\partial \mathbf{n}} \left(\mathbf{x} \right) \varepsilon^2 \mathrm{sin}\theta \mathrm{d}\theta \mathrm{d}\varphi \right)$$

where it is clear that

$$\lim_{\varepsilon \to 0} \left(\int_{\theta,\varphi} \frac{1}{4\pi\varepsilon} \frac{\partial \phi}{\partial \mathbf{n}} \left(\mathbf{x} \right) \varepsilon^2 \mathrm{sin}\theta \mathrm{d}\theta \mathrm{d}\varphi \right) \longrightarrow 0 \qquad \boxed{\text{A.2}}$$

If we analyse the behaviour of the integral kernel

$$\frac{1}{4\pi r} \frac{\partial \phi}{\partial \mathbf{n}} \left(\mathbf{x} \right) \qquad \boxed{\text{A.3}}$$

when $\mathbf{x} \longrightarrow \mathbf{z}$ it is clear that $w^* \sim \frac{1}{r}$, and remembering the assumption of smoothness of the boundary at $\mathbf{z}$, $\mathrm{d}\Gamma \sim r\mathrm{d}r\mathrm{d}\theta$ so we can conclude that

$$\fint_\Gamma \frac{1}{4\pi r} \frac{\partial \phi}{\partial \mathbf{n}} \left(\mathbf{x} \right) \mathrm{d}\Gamma = \int_\Gamma \frac{1}{4\pi r} \frac{\partial \phi}{\partial \mathbf{n}} \left(\mathbf{x} \right) \mathrm{d}\Gamma \longrightarrow \text{weakly singular} \qquad \boxed{\text{A.4}}$$

so the former integral can be calculated directly as an improper one and it is not necessary to be defined as a principal value.

Respectively

$$\int_{\Gamma} \frac{\partial w^*}{\partial \mathbf{n}} (\mathbf{x}; \mathbf{z}) \phi(\mathbf{x}) \, d\Gamma = - \int_{\Gamma} \frac{r_{,i} n_i}{4\pi r^2} \phi(\mathbf{x}) \, d\Gamma =$$

$$-\lim_{\varepsilon \to 0} \left(\int_{\Gamma - \Gamma_\varepsilon} \frac{r_{,i} n_i}{4\pi r^2} \phi(\mathbf{x}) \, d\Gamma \right) - \lim_{\varepsilon \to 0} \left(\int_{\Gamma_\varepsilon} \frac{r_{,i} n_i}{4\pi r^2} \phi(\mathbf{x}) \, d\Gamma \right) = \qquad \boxed{\text{A.5}}$$

$$-\int_{\Gamma} \frac{r_{,i} n_i}{4\pi^2} \phi(\mathbf{x}) \, d\Gamma - \lim_{\varepsilon \to 0} \left(\int_{\theta,\varphi} \frac{r_{,i} n_i}{4\pi \varepsilon^2} \phi(\mathbf{x}) \varepsilon^2 \sin\theta \, d\theta \, d\varphi \right)$$

Taking into account that $\mathbf{n}$ and $\mathbf{r}$ are parallel in Γ_ε then $r_{,i} n_i = \dfrac{r_i n_i}{r} = 1$ so

$$\lim_{\varepsilon \to 0} \left(\int_{\theta,\varphi} \frac{r_{,i} n_i}{4\pi \varepsilon^2} \phi(\mathbf{x}) \varepsilon^2 \sin\theta \, d\theta \, d\varphi \right) = \int_{\theta,\varphi} \frac{\phi(\mathbf{z})}{4\pi} \sin\theta \, d\theta \, d\varphi = \frac{\phi(\mathbf{z})}{4\pi} \Delta\Omega(\mathbf{z}) \qquad \boxed{\text{A.6}}$$

where $\Delta\Omega(\mathbf{z})$ is the solid angle centred in $\mathbf{z}$ (2π for a smooth boundary point)

Again when $\mathbf{x} \longrightarrow \mathbf{z}$ we can state[1] that $\dfrac{\partial w^*}{\partial \mathbf{n}} \sim \dfrac{1}{r}$, and remembering the assumption of smoothness of the boundary at $\mathbf{z}$, $d\Gamma \sim r \, dr \, d\theta$ we reach to

$$\fint_{\Gamma} \frac{r_{,i} n_i}{4\pi r^2} \phi(\mathbf{x}) \, d\Gamma = \int_{\Gamma} \frac{r_{,i} n_i}{4\pi r^2} \phi(\mathbf{x}) \, d\Gamma \longrightarrow \text{weakly singular} \qquad \boxed{\text{A.7}}$$

---

[1]It can be proved that the first order term depends on the curvatures of the surface.

A.2   Derivatives of the BIE associated with the Laplace operator

A.2.1   3D Domains

General expressions

$$r = \| \, \mathbf{x} - \mathbf{z} \, \| \qquad\qquad r_i = x_i - z_i$$

$$\frac{\partial r}{\partial x_i} = r_{,i} = \frac{r_i}{r} \qquad \frac{\partial r_{,i}}{\partial x_j} = r_{,ij} = \frac{\delta_{ij} - r_{,i} r_{,j}}{r}$$

(A.8)

It is easy to verify that

$$\frac{\partial [f(r)]}{\partial z_i} = -\frac{\partial [f(r)]}{\partial x_i}$$

(A.9)

Fundamental solution for 3D Laplace operator

$$w^* = \frac{1}{4\pi r} \qquad \frac{\partial w^*}{\partial z_i} = \frac{r_{,i}}{4\pi r^2} \qquad \frac{\partial^2 w^*}{\partial z_i \partial z_j} = \frac{3 r_{,i} r_{,j} - \delta_{ij}}{4\pi r^3}$$

(A.10)

$$q^* = \frac{\partial w^*}{\partial x_i} n_i = \frac{\partial w^*}{\partial \mathbf{n}} = -\frac{r_{,i} n_i}{4\pi r^2} = -\frac{r_{,n}}{4\pi r^2} \qquad \frac{\partial q^*}{\partial z_i} = \frac{n_i - 3 r_{,i} r_{,n}}{4\pi r^3}$$

(A.11)

u-BIE

The standard boundary integral equation in the case of a scalar problem can be written as

$$\phi(\mathbf{z}) + \int_{\Gamma} q^*(\mathbf{x}; \mathbf{z}) \, \phi(\mathbf{x}) \, \mathrm{d}\Gamma = \int_{\Gamma} w^*(\mathbf{x}; \mathbf{z}) \, q(\mathbf{x}) \, \mathrm{d}\Gamma$$

(A.12)

where

$$q(\mathbf{x}) = \frac{\partial \phi(\mathbf{x})}{\partial \mathbf{n}} \tag{A.13}$$

If we differentiate equation (A.12) along the z coordinate we obtain

$$\phi_{,i}(\mathbf{z}) + \int_{\Gamma} \frac{\partial q^*(\mathbf{x};\mathbf{z})}{\partial z_i} \phi(\mathbf{x}) \, d\Gamma = \int_{\Gamma} \frac{\partial w^*(\mathbf{x};\mathbf{z})}{\partial z_i} q(\mathbf{x}) \, d\Gamma \tag{A.14}$$

It must be pointed that it is assumed that $\mathbf{z}$ is a smooth boundary point in the following steps in which the previous integral will be analysed.

The next step is to introduce a slight modification in the domain as shown in figure (A.1) and apply equation (A.14) to this domain to get

$$\phi_{,i}(\mathbf{z}) + \int_{(\Gamma-\Gamma_\varepsilon)+\Gamma_\varepsilon} \frac{\partial q^*(\mathbf{x};\mathbf{z})}{\partial z_i} \phi(\mathbf{x}) \, d\Gamma = \int_{(\Gamma-\Gamma_\varepsilon)+\Gamma_\varepsilon} \frac{\partial w^*(\mathbf{x};\mathbf{z})}{\partial z_i} q(\mathbf{x}) \, d\Gamma \tag{A.15}$$

If we particularize the previous equation using (A.10), (A.11)

$$\phi_{,i}(\mathbf{z}) + \overbrace{\int_{\Gamma_\varepsilon} \frac{n_i(\mathbf{x})}{4\pi r^3} \phi(\mathbf{x}) \, d\Gamma}^{\text{T}_1} - \overbrace{\int_{\Gamma_\varepsilon} \frac{3r_{,i}r_{,n}}{4\pi r^3} \phi(\mathbf{x}) \, d\Gamma}^{\text{T}_2} - \overbrace{\int_{\Gamma_\varepsilon} \frac{r_{,i}}{4\pi r^2} q(\mathbf{x}) \, d\Gamma}^{\text{T}_3} + \tag{A.16}$$

$$\overbrace{\int_{\Gamma-\Gamma_\varepsilon} \frac{n_i(\mathbf{x})}{4\pi r^3} \phi(\mathbf{x}) \, d\Gamma}^{\text{T}_4} - \overbrace{\int_{\Gamma-\Gamma_\varepsilon} \frac{3r_{,i}r_{,n}}{4\pi r^3} \phi(\mathbf{x}) \, d\Gamma}^{\text{T}_5} - \overbrace{\int_{\Gamma-\Gamma_\varepsilon} \frac{r_{,i}}{4\pi r^2} q(\mathbf{x}) \, d\Gamma}^{\text{T}_6} = 0$$

where the different integrals involved have been labelled in order to analyse them sequentially.

In what follows the main variables $\phi(\mathbf{x})$ and $q(\mathbf{x})$ are approximated using a Taylor expansion centred in $\mathbf{z}$. Taking this into account we can write

$$\phi\left(\mathbf{x}\right) \approx \phi\left(\mathbf{z}\right) + O\left(r\right)$$

$$\phi\left(\mathbf{x}\right) \approx \phi\left(\mathbf{z}\right) + \phi_{,k}\left(\mathbf{z}\right) r_{,k} + O\left(r^2\right) \tag{A.17}$$

$$q\left(\mathbf{x}\right) = \phi_{,k}\left(\mathbf{x}\right) n_k\left(\mathbf{x}\right) \approx \phi_{,k}\left(\mathbf{z}\right) n_{,k}\left(\mathbf{x}\right) + O\left(r\right)$$

In the case of the integrals over Γ_ε it is straightforward to check that

- r is constant
- $r_{,i} = n_i \Longrightarrow r_{,n} = 1$
- $d\Gamma = r^2 \sin\theta d\theta d\varphi$

taking this into account we can write

Term $\boxed{\mathrm{T}_1 \text{ - } \mathrm{T}_2}$

$$\int_{\Gamma_\varepsilon} \frac{n_i\left(\mathbf{x}\right) - 3r_{,i}r_{,n}}{4\pi r^3} \phi\left(\mathbf{x}\right) d\Gamma = \overbrace{\int_{\Gamma_\varepsilon} \frac{n_i\left(\mathbf{x}\right) - 3r_{,i}r_{,n}}{4\pi r^3} \left[\phi\left(\mathbf{x}\right) - \phi\left(\mathbf{z}\right) - \phi_{,k}\left(\mathbf{z}\right) r_k\right] d\Gamma}^{\displaystyle \lim_{\varepsilon \to 0} \longrightarrow 0}$$

$$+ \phi\left(\mathbf{z}\right) \overbrace{\int_{\Gamma_\varepsilon} \frac{n_i\left(\mathbf{x}\right) - 3r_{,i}r_{,n}}{4\pi r^3} d\Gamma}^{\mathrm{A}_1} + \phi_{,k}\left(\mathbf{z}\right) \overbrace{\int_{\Gamma_\varepsilon} \frac{n_i\left(\mathbf{x}\right) r_{,k} - 3r_{,i}r_{,n}r_{,k}}{4\pi r^2} d\Gamma}^{\mathrm{A}_2} \tag{A.18}$$

In order to analyse A_1 we can assume without loss of generality that $\mathbf{n}\left(\mathbf{z}\right) = \mathbf{e}_3$.

Term $\boxed{\mathrm{A}_1}$

$$\phi\left(\mathbf{z}\right) \int_{\Gamma_\varepsilon} \frac{n_i\left(\mathbf{x}\right) - 3r_{,i}r_{,n}}{4\pi r^3} d\Gamma = -\phi\left(\mathbf{z}\right) \int_{\Gamma_\varepsilon} \frac{n_i\left(\mathbf{x}\right)}{2\pi r^3} d\Gamma \tag{A.19}$$

so, taking into account that $n_i = r_{,i}$, the intervals of integration of the hemisphere are respectively $\theta\left[0 \to \pi/2\right]$ and $\varphi\left[0 \to 2\pi\right]$ and the components of the normal vector can be expressed as

$$n_1 = \sin\theta\cos\varphi \qquad n_2 = \sin\theta\sin\varphi \qquad n_3 = \cos\theta \qquad \text{(A.20)}$$

so A_1 can be solved analytically.

$$-\phi\left(\mathbf{z}\right)\int_{\Gamma_\varepsilon}\frac{n_i\left(\mathbf{x}\right)}{2\pi r^3}\mathrm{d}\Gamma = -\phi\left(\mathbf{z}\right)\left(\int_{\theta,\varphi}\frac{n_i\left(\mathbf{x}\right)}{2\pi\varepsilon^3}\varepsilon^2\sin\theta\mathrm{d}\theta\mathrm{d}\varphi\right) =$$

$$-\phi\left(\mathbf{z}\right)\int_{\theta,\varphi}\frac{n_i\left(\mathbf{x}\right)}{2\pi\varepsilon}\sin\theta\mathrm{d}\theta\mathrm{d}\varphi = -\overbrace{\frac{\phi\left(\mathbf{z}\right)}{2\varepsilon}}^{\text{I}_1}\delta_{i3} \qquad \text{(A.21)}$$

At the same time

Term $\boxed{-\text{T}_3}$

$$-\int_{\Gamma_\varepsilon}\frac{r_{,i}}{4\pi r^2}q\left(\mathbf{x}\right)\mathrm{d}\Gamma = -\overbrace{\int_{\Gamma_\varepsilon}\frac{r_{,i}}{4\pi r^2}\left[q\left(\mathbf{x}\right)-\phi_{,k}\left(\mathbf{z}\right)n_k\left(\mathbf{x}\right)\right]\mathrm{d}\Gamma}^{\displaystyle\lim_{\varepsilon\to 0}\longrightarrow 0}$$

$$-\overbrace{\phi_{,k}\left(\mathbf{z}\right)\int_{\Gamma_\varepsilon}\frac{r_{,i}n_k\left(\mathbf{x}\right)}{4\pi r^2}\mathrm{d}\Gamma}^{\text{A}_3} \qquad \text{(A.22)}$$

so if we combine A_2 and A_3 and we apply the previous remarks then we get

APPENDIX A. KERNELS AND LIMITS

Term $\boxed{A_2 - A_3}$

$$\phi_{,k}\left(\mathbf{z}\right)\int_{\Gamma_\varepsilon}\frac{n_i\left(\mathbf{x}\right)r_{,k}-3r_{,i}r_{,n}r_{,k}-n_k\left(\mathbf{x}\right)r_{,i}}{4\pi r^2}\mathrm{d}\Gamma=-\phi_{,k}\left(\mathbf{z}\right)\int_{\Gamma_\varepsilon}\frac{3n_i\left(\mathbf{x}\right)n_k\left(\mathbf{x}\right)}{4\pi r^2}\mathrm{d}\Gamma=$$

$$-\phi_{,k}\left(\mathbf{z}\right)\int_{\theta,\varphi}\frac{3n_i\left(\mathbf{x}\right)n_k\left(\mathbf{x}\right)}{4\pi\varepsilon^2}\varepsilon^2\sin\theta\mathrm{d}\theta\mathrm{d}\varphi=-\phi_{,k}\left(\mathbf{z}\right)\int_{\theta,\varphi}\frac{3n_i\left(\mathbf{x}\right)n_k\left(\mathbf{x}\right)}{4\pi}\sin\theta\mathrm{d}\theta\mathrm{d}\varphi=$$

$$-\phi_{,k}\left(\mathbf{z}\right)\frac{\delta_{ik}}{2}=-\frac{\phi_{,i}\left(\mathbf{z}\right)}{2}$$

$$\boxed{\text{A.23}}$$

In the case of the integrals over $\Gamma - \Gamma_\varepsilon$ it is also necessary to analyse the behaviour close to the singularity. In this case (assuming smooth boundary), $d\Gamma \sim r d\theta dr$.

Term $\boxed{T_4}$

$$\int_{\Gamma-\Gamma_\varepsilon}\frac{n_i\left(\mathbf{x}\right)}{4\pi r^3}\phi\left(\mathbf{x}\right)\mathrm{d}\Gamma=\overbrace{\int_{\Gamma-\Gamma_\varepsilon}\frac{n_i\left(\mathbf{x}\right)}{4\pi r^3}\left[\phi\left(\mathbf{x}\right)-\phi\left(\mathbf{z}\right)-\phi_{,k}\left(\mathbf{z}\right)r_k\right]\mathrm{d}\Gamma}^{A_4}$$

$$\boxed{\text{A.24}}$$

$$+\overbrace{\phi\left(\mathbf{z}\right)\int_{\Gamma-\Gamma_\varepsilon}\frac{n_i\left(\mathbf{x}\right)}{4\pi r^3}\mathrm{d}\Gamma}^{A_5}+\overbrace{\phi_{,k}\left(\mathbf{z}\right)\int_{\Gamma-\Gamma_\varepsilon}\frac{n_i\left(\mathbf{x}\right)r_k}{4\pi r^3}\mathrm{d}\Gamma}^{E_1}$$

Remembering that

$$\boldsymbol{\nabla}_s\phi=\boldsymbol{\nabla}\phi-\mathbf{n}\left(\mathbf{n}\bullet\boldsymbol{\nabla}\phi\right)\Longrightarrow\phi_{,k}=\phi_{,k}^{\mathcal{S}}+n_k q\qquad\boxed{\text{A.25}}$$

$$\lim_{\varepsilon\to0}\left(\int_{\Gamma-\Gamma_\varepsilon}\mathrm{I}\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma\right)=\fint_\Gamma\mathrm{I}\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma\qquad\boxed{\text{A.26}}$$

and taking into account that if $\mathrm{I}\left(\mathbf{x};\mathbf{z}\right)$ is weakly singular in Γ

$$\fint_\Gamma I(\mathbf{x}; \mathbf{z}) \, d\Gamma = \int_\Gamma I(\mathbf{x}; \mathbf{z}) \, d\Gamma \qquad \boxed{\text{A.27}}$$

Term $\boxed{A_4}$

$$\int_{\Gamma - \Gamma_\varepsilon} \frac{n_i(\mathbf{x})}{4\pi r^3} \left[\phi(\mathbf{x}) - \phi(\mathbf{z}) - \phi_{,k}(\mathbf{z}) r_k \right] d\Gamma =$$

$$\boxed{\text{A.28}}$$

$$\overbrace{\int_\Gamma \frac{n_i(\mathbf{x})}{4\pi r^3} \left[\phi(\mathbf{x}) - \phi(\mathbf{z}) - \phi_{,k}^s(\mathbf{z}) r_k \right] d\Gamma}^{\text{weakly singular}} - \overbrace{n_k(\mathbf{z}) q(\mathbf{z}) \int_\Gamma \frac{n_i(\mathbf{x}) r_{,k}}{4\pi r^2} d\Gamma}^{\text{weakly singular}}$$

where it has been taking into account that $n_k(\mathbf{z}) r_{,k} \to 0$ for $x \to z$.

By means of Stokes Theorem

Term $\boxed{A_5}$

$$\phi(\mathbf{z}) \int_{\Gamma - \Gamma_\varepsilon} \frac{n_i(\mathbf{x})}{4\pi r^3} d\Gamma = \phi(\mathbf{z}) \int_{\Gamma - \Gamma_\varepsilon} \left[\frac{n_i(\mathbf{x})}{4\pi r^3} - \frac{3r_{,i} r_{,n}}{4\pi r^3} + \frac{3r_{,i} r_{,n}}{4\pi r^3} \right] d\Gamma =$$

$$\boxed{\text{A.29}}$$

$$\overbrace{\phi(\mathbf{z}) \int_{\Gamma - \Gamma_\varepsilon} \frac{3r_{,i} r_{,n}}{4\pi r^3} d\Gamma}^{E_2} + \overbrace{\phi(\mathbf{z}) \int_{\Gamma - \Gamma_\varepsilon} \left[\nabla \wedge \frac{\mathbf{r} \wedge \mathbf{e}_i}{4\pi r^3} \right] \mathbf{n}(\mathbf{x}) \, d\Gamma}^{A_6}$$

Term $\boxed{A_6}$

$$\phi(\mathbf{z}) \int_{\Gamma - \Gamma_\varepsilon} \left[\nabla \wedge \frac{\mathbf{r} \wedge \mathbf{e}_i}{4\pi r^3} \right] \mathbf{n}(\mathbf{x}) \, d\Gamma = \phi(\mathbf{z}) \oint_{\Gamma - \Gamma_\varepsilon} \frac{\mathbf{r} \wedge \mathbf{e}_i}{4\pi r^3} d\mathbf{l} =$$

$$\boxed{\text{A.30}}$$

$$\overbrace{\phi(\mathbf{z}) \oint_\Gamma \frac{\mathbf{r} \wedge \mathbf{e}_i}{4\pi r^3} d\mathbf{l}}^{\text{regular}} + \overbrace{\phi(\mathbf{z}) \oint_{\Gamma_\varepsilon} \frac{\mathbf{r} \wedge \mathbf{e}_i}{4\pi r^3} d\mathbf{l}}^{A_7}$$

using again the assumption $n(\mathbf{z}) = \mathbf{e}_3$ it can be easily proved that

$$\lim_{\varepsilon \to 0} d\mathbf{l} \longrightarrow \left[\frac{\mathbf{r}}{r} \wedge \mathbf{e}_3 \right] r d\theta = [\mathbf{r} \wedge \mathbf{e}_3] \, d\theta \qquad (A.31)$$

and if we introduce the previous expression we get

Term $\boxed{A_7}$

$$\phi(\mathbf{z}) \oint_{\Gamma_\varepsilon} \frac{\mathbf{r} \wedge \mathbf{e}_i}{4\pi r^3} d\mathbf{l} = \phi(\mathbf{z}) \int_\theta \frac{\mathbf{r} \wedge \mathbf{e}_i}{4\pi r^3} [\mathbf{r} \wedge \mathbf{e}_3] \, d\theta = \overbrace{\frac{\phi(\mathbf{z})}{2\varepsilon}}^{I_1} \delta_{i3} \qquad (A.32)$$

where the unbounded term I_1 can be cancelled with the corresponding term from equation (A.21).

If we expand T_5 taking into account that $r_{,n} \to 0$ for $x \to z$ we get

Term $\boxed{-T_5}$

$$-\int_{\Gamma - \Gamma_\varepsilon} \frac{3r_{,i}r_{,n}}{4\pi r^3} \phi(\mathbf{x}) \, d\Gamma = -\overbrace{\int_\Gamma \frac{3r_{,i}r_{,n}}{4\pi r^3} [\phi(\mathbf{x}) - \phi(\mathbf{z})] \, d\Gamma}^{\text{weakly singular}}$$

$$-\phi(\mathbf{z}) \overbrace{\int_{\Gamma - \Gamma_\varepsilon} \frac{3r_{,i}r_{,n}}{4\pi r^3} d\Gamma}^{E_2} \qquad (A.33)$$

where we obtain a term labelled again as E_2, that cancels the E_2 term from equation (A.29).

If we analyse the last terms

Term $\boxed{\text{-T}_6}$

$$-\int_{\Gamma-\Gamma_\varepsilon} \frac{r_{,i}}{4\pi r^2} q\left(\mathbf{x}\right) \mathrm{d}\Gamma = -\overbrace{\int_\Gamma \frac{r_{,i}}{4\pi r^2} \left[q\left(\mathbf{x}\right) - \phi_{,k}\left(\mathbf{z}\right) n_k\left(\mathbf{x}\right)\right] \mathrm{d}\Gamma}^{\text{A}_8}$$

$$-\phi_{,k}\left(\mathbf{z}\right) \overbrace{\int_{\Gamma-\Gamma_\varepsilon} \frac{r_{,i} n_k\left(\mathbf{x}\right)}{4\pi r^2} \mathrm{d}\Gamma}^{\text{A}_9}$$

(A.34)

Term $\boxed{\text{-A}_8}$

$$-\int_\Gamma \frac{r_{,i}}{4\pi r^2} \left[q\left(\mathbf{x}\right) - \phi_{,k}\left(\mathbf{z}\right) n_k\left(\mathbf{x}\right)\right] \mathrm{d}\Gamma = -\overbrace{\int_\Gamma \frac{r_{,i}}{4\pi r^2} \left[q\left(\mathbf{x}\right) - q\left(\mathbf{z}\right)\right] \mathrm{d}\Gamma}^{\text{weakly singular}}$$

(A.35)

$$-\phi_{,k}\left(\mathbf{z}\right) \overbrace{\int_\Gamma \frac{r_{,i}}{4\pi r^2} \left[n_k\left(\mathbf{z}\right) - n_k\left(\mathbf{x}\right)\right] \mathrm{d}\Gamma}^{\text{weakly singular}}$$

Term $\boxed{\text{-A}_9}$

$$-\phi_{,k}\left(\mathbf{z}\right) \int_{\Gamma-\Gamma_\varepsilon} \frac{r_{,i} n_k\left(\mathbf{x}\right)}{4\pi r^2} \mathrm{d}\Gamma = -\phi_{,k}\left(\mathbf{z}\right) \overbrace{\int_{\Gamma-\Gamma_\varepsilon} \frac{r_{,k} n_i\left(\mathbf{x}\right)}{4\pi r^2} \mathrm{d}\Gamma}^{\text{E}_1}$$

(A.36)

$$-\phi_{,k}\left(\mathbf{z}\right) \overbrace{\oint_{\Gamma-\Gamma_\varepsilon} \boldsymbol{\nabla} \wedge \left[\frac{\mathbf{e}_i \wedge \mathbf{e}_k}{4\pi r}\right] \mathbf{n}\left(\mathbf{x}\right) \mathrm{d}\Gamma}^{\text{A}_{10}}$$

where we obtain a term labelled again as E_1, that cancels the E_1 term from equation (A.24).

APPENDIX A. KERNELS AND LIMITS

Term $\boxed{\text{-A}_{10}}$

$$-\phi_{,k}\left(\mathbf{z}\right)\oint_{\Gamma-\Gamma_\varepsilon}\boldsymbol{\nabla}\wedge\left[\frac{\mathbf{e}_i\wedge\mathbf{e}_k}{4\pi r}\right]\mathbf{n}\left(\mathbf{x}\right)\mathrm{d}\Gamma=\overbrace{-\phi_{,k}\left(\mathbf{z}\right)\oint_\Gamma\frac{\mathbf{e}_i\wedge\mathbf{e}_k}{4\pi r}\mathrm{d}\mathbf{l}}^{\text{regular}}$$

$$\overbrace{-\phi_{,k}\left(\mathbf{z}\right)\oint_{\Gamma_\varepsilon}\frac{\mathbf{e}_i\wedge\mathbf{e}_k}{4\pi r}\mathrm{d}\mathbf{l}}^{\text{A}_{11}}$$

$$\text{(A.37)}$$

Term $\boxed{\text{A}_{11}}$

$$\phi_{,k}\left(\mathbf{z}\right)\oint_{\Gamma_\varepsilon}\frac{\mathbf{e}_i\wedge\mathbf{e}_k}{4\pi r}\mathrm{d}\mathbf{l}=\phi_{,k}\left(\mathbf{z}\right)\int_\theta\frac{\mathbf{e}_i\wedge\mathbf{e}_k}{4\pi}\left[\frac{\mathbf{r}}{r}\wedge\mathbf{e}_3\right]\mathrm{d}\theta\qquad\text{(A.38)}$$

By symmetry it is easy to check that

$$\lim_{\varepsilon\to0}\left(\phi_{,k}\left(\mathbf{z}\right)\int_\theta\frac{\mathbf{e}_i\wedge\mathbf{e}_k}{4\pi}\left[\frac{\mathbf{r}}{r}\wedge\mathbf{e}_3\right]\mathrm{d}\theta\right)\longrightarrow0\qquad\text{(A.39)}$$

so reordering the terms we have

$$\frac{1}{2}\phi_{,i}\left(\mathbf{z}\right)-\phi_{,k}\left(\mathbf{z}\right)\int_\Gamma\frac{r_{,i}}{4\pi r^2}\left[n_k\left(\mathbf{z}\right)-n_k\left(\mathbf{x}\right)\right]\mathrm{d}\Gamma-\phi_{,k}\left(\mathbf{z}\right)\oint_\Gamma\frac{\mathbf{e}_i\wedge\mathbf{e}_k}{4\pi r}\mathrm{d}\mathbf{l}+$$

$$\int_\Gamma\frac{n_i\left(\mathbf{x}\right)}{4\pi r^3}\left[\phi\left(\mathbf{x}\right)-\phi\left(\mathbf{z}\right)-\phi_{,k}^s\left(\mathbf{z}\right)r_k\right]\mathrm{d}\Gamma-\int_\Gamma\frac{3r_{,i}r_{,n}}{4\pi r^3}\left[\phi\left(\mathbf{x}\right)-\phi\left(\mathbf{z}\right)\right]\mathrm{d}\Gamma+\quad\text{(A.40)}$$

$$\phi\left(\mathbf{z}\right)\oint_\Gamma\frac{\mathbf{r}\wedge\mathbf{e}_i}{4\pi r^3}\mathrm{d}\mathbf{l}=\int_\Gamma\frac{r_{,i}}{4\pi r^2}\left[q\left(\mathbf{x}\right)-q\left(\mathbf{z}\right)\right]\mathrm{d}\Gamma+n_k\left(\mathbf{z}\right)q\left(\mathbf{z}\right)\int_\Gamma\frac{n_i\left(\mathbf{x}\right)r_{,k}}{4\pi r^2}\mathrm{d}\Gamma$$

A.2.2   2D Domains

General expressions

$$r = \| \mathbf{x} - \mathbf{z} \| \qquad\qquad r_i = x_i - z_i$$

$$\frac{\partial r}{\partial x_i} = r_{,i} = \frac{r_i}{r} \qquad \frac{\partial r_{,i}}{\partial x_j} = r_{,ij} = \frac{\delta_{ij} - r_{,i} r_{,j}}{r}$$

<div align="right">(A.41)</div>

It is easy to verify that

$$\frac{\partial [f(r)]}{\partial z_i} = -\frac{\partial [f(r)]}{\partial x_i}$$

<div align="right">(A.42)</div>

Fundamental solution for 2D Laplace operator

$$w^* = -\frac{1}{2\pi} \log r \qquad \frac{\partial w^*}{\partial z_i} = \frac{r_{,i}}{2\pi r} \qquad \frac{\partial^2 w^*}{\partial z_i \partial z_j} = \frac{2 r_{,i} r_{,j} - \delta_{ij}}{2\pi r^2}$$

<div align="right">(A.43)</div>

$$q^* = \frac{\partial w^*}{\partial x_i} n_i = \frac{\partial w^*}{\partial \mathbf{n}} = -\frac{r_{,i} n_i}{4\pi r^2} = -\frac{r_{,n}}{4\pi r^2} \qquad \frac{\partial q^*}{\partial z_i} = \frac{2 r_{,i} r_{,n} - n_i}{2\pi r^2}$$

<div align="right">(A.44)</div>

u-BIE

The standard boundary integral equation in the case of a scalar problem can be written as

$$\phi(\mathbf{z}) + \int_\Gamma q^*(\mathbf{x}; \mathbf{z}) \, \phi(\mathbf{x}) \, \mathrm{d}\Gamma = \int_\Gamma w^*(\mathbf{x}; \mathbf{z}) \, q(\mathbf{x}) \, \mathrm{d}\Gamma$$

<div align="right">(A.45)</div>

where

$$q(\mathbf{x}) = \frac{\partial \phi(\mathbf{x})}{\partial \mathbf{n}}$$

<div align="right">(A.46)</div>

If we differentiate equation (A.45) along the z coordinate we obtain

$$\phi_{,i}(\mathbf{z}) + \int_{\Gamma} \frac{\partial q^*(\mathbf{x};\mathbf{z})}{\partial z_i} \phi(\mathbf{x}) \, d\Gamma = \int_{\Gamma} \frac{\partial w^*(\mathbf{x};\mathbf{z})}{\partial z_i} q(\mathbf{x}) \, d\Gamma \qquad \boxed{\text{A.47}}$$

It must be pointed that it is assumed that $\mathbf{z}$ is a smooth boundary point in the following steps in which the previous integral will be analysed.
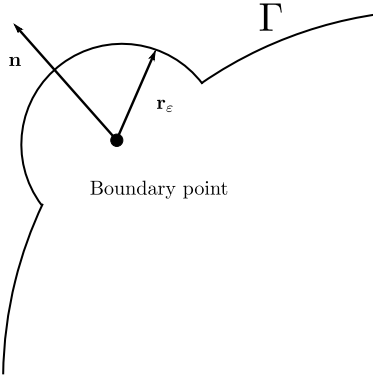
Figure A.2: semicircle around a boundary point at $\mathbf{z}$

The next step is to introduce a slight modification in the domain as shown in figure (A.2) and apply equation (A.47) to this domain to get

$$\phi_{,i}(\mathbf{z}) + \int_{(\Gamma-\Gamma_\varepsilon)+\Gamma_\varepsilon} \frac{\partial q^*(\mathbf{x};\mathbf{z})}{\partial z_i} \phi(\mathbf{x}) \, d\Gamma = \int_{(\Gamma-\Gamma_\varepsilon)+\Gamma_\varepsilon} \frac{\partial w^*(\mathbf{x};\mathbf{z})}{\partial z_i} q(\mathbf{x}) \, d\Gamma \qquad \boxed{\text{A.48}}$$

If we particularize the previous equation using equations (A.43), (A.44)

$$\phi_{,i}\left(\mathbf{z}\right) + \overbrace{\int_{\Gamma_{\varepsilon}} \frac{n_i\left(\mathbf{x}\right)}{2\pi r^2}\phi\left(\mathbf{x}\right)\mathrm{d}\Gamma}^{T_1} - \overbrace{\int_{\Gamma_{\varepsilon}} \frac{2r_{,i}r_{,n}}{2\pi r^2}\phi\left(\mathbf{x}\right)\mathrm{d}\Gamma}^{T_2} - \overbrace{\int_{\Gamma_{\varepsilon}} \frac{r_{,i}}{2\pi r}q\left(\mathbf{x}\right)\mathrm{d}\Gamma}^{T_3} +$$

$$\overbrace{\int_{\Gamma-\Gamma_{\varepsilon}} \frac{n_i\left(\mathbf{x}\right)}{2\pi r^2}\phi\left(\mathbf{x}\right)\mathrm{d}\Gamma}^{T_4} - \overbrace{\int_{\Gamma-\Gamma_{\varepsilon}} \frac{2r_{,i}r_{,n}}{2\pi r^2}\phi\left(\mathbf{x}\right)\mathrm{d}\Gamma}^{T_5} - \overbrace{\int_{\Gamma-\Gamma_{\varepsilon}} \frac{r_{,i}}{2\pi r}q\left(\mathbf{x}\right)\mathrm{d}\Gamma}^{T_6} = 0$$

(A.49)

where the different integrals involved have been labelled in order to analyse them sequentially.

In what follows the main variables $\phi\left(\mathbf{x}\right)$ and $q\left(\mathbf{x}\right)$ will be approximated using a Taylor expansion centered in $\mathbf{z}$. Taking this into account we can write

$$\phi\left(\mathbf{x}\right) \approx \phi\left(\mathbf{z}\right) + O\left(r\right)$$

$$\phi\left(\mathbf{x}\right) \approx \phi\left(\mathbf{z}\right) + \phi_{,k}\left(\mathbf{z}\right)r_{,k} + O\left(r^2\right)$$

(A.50)

$$q\left(\mathbf{x}\right) = \phi_{,k}\left(\mathbf{x}\right)n_k\left(\mathbf{x}\right) \approx \phi_{,k}\left(\mathbf{z}\right)n_{,k}\left(\mathbf{x}\right) + O\left(r\right)$$

In the case of integrals over Γ_{ε} it is straightforward to check that

- r is constant

- $r_{,i} = n_i \Longrightarrow r_{,n} = 1$

- $\mathrm{d}\Gamma = r\mathrm{d}\theta$

taking this into account we can write

Term $\boxed{T_1 \text{ - } T_2}$

$$\int_{\Gamma_\varepsilon} \frac{n_i(\mathbf{x}) - 2r_{,i}r_{,n}}{2\pi r^2} \phi(\mathbf{x})\, d\Gamma = \overbrace{\int_{\Gamma_\varepsilon} \frac{n_i(\mathbf{x}) - 2r_{,i}r_{,n}}{2\pi r^2} \left[\phi(\mathbf{x}) - \phi(\mathbf{z}) - \phi_{,k}(\mathbf{z})\, r_k\right] d\Gamma}^{\displaystyle \lim_{\varepsilon \to 0} \longrightarrow 0}$$

$$+ \phi(\mathbf{z}) \overbrace{\int_{\Gamma_\varepsilon} \frac{n_i(\mathbf{x}) - 2r_{,i}r_{,n}}{2\pi r^2} d\Gamma}^{A_1} + \phi_{,k}(\mathbf{z}) \overbrace{\int_{\Gamma_\varepsilon} \frac{n_i(\mathbf{x})\, r_{,k} - 2r_{,i}r_{,n}r_{,k}}{2\pi r} d\Gamma}^{A_2} \qquad \boxed{\text{A.51}}$$

If we analyse A_1

Term $\boxed{A_1}$

$$\phi(\mathbf{z}) \int_{\Gamma_\varepsilon} \frac{n_i(\mathbf{x}) - 2r_{,i}r_{,n}}{2\pi r^2} d\Gamma = -\phi(\mathbf{z}) \int_{\Gamma_\varepsilon} \frac{n_i(\mathbf{x})}{2\pi r^2} d\Gamma \qquad \boxed{\text{A.52}}$$

Taking into account that $n_i = r_{,i}$, the interval of integration of the semicircle is $\theta\,[0 \to \pi]$ and the components of the normal vector can be expressed as

$$n_1 = \cos\theta \qquad n_2 = \sin\theta \qquad \boxed{\text{A.53}}$$

so A_1 can be solved analytically.

$$-\phi(\mathbf{z}) \lim_{\varepsilon \to 0} \left(\int_\theta \frac{n_i(\mathbf{x})}{2\pi\varepsilon^2} \varepsilon\, d\theta \right) = -\phi(\mathbf{z}) \int_\theta \frac{n_i(\mathbf{x})}{2\pi\varepsilon} d\theta = -\overbrace{\frac{\phi(\mathbf{z})\, n_i(\mathbf{z})}{\pi\varepsilon}}^{I_1} \qquad \boxed{\text{A.54}}$$

At the same time

Term $\boxed{-T_3}$

$$-\int_{\Gamma_\varepsilon} \frac{r_{,i}}{2\pi r} q\left(\mathbf{x}\right) d\Gamma = -\overbrace{\int_{\Gamma_\varepsilon} \frac{r_{,i}}{2\pi r} \left[q\left(\mathbf{x}\right) - \phi_{,k}\left(\mathbf{z}\right) n_k\left(\mathbf{x}\right)\right] d\Gamma}^{\underbrace{\lim_{\varepsilon \to 0} \longrightarrow 0}}$$

$$\text{(A.55)}$$

$$-\phi_{,k}\left(\mathbf{z}\right) \overbrace{\int_{\Gamma_\varepsilon} \frac{r_{,i} n_k\left(\mathbf{x}\right)}{2\pi r} d\Gamma}^{A_3}$$

so if we combine A_2 and A_3 applying the previous remarks we get

Term $\boxed{A_2 \text{ - } A_3}$

$$\phi_{,k}\left(\mathbf{z}\right) \int_{\Gamma_\varepsilon} \frac{n_i\left(\mathbf{x}\right) r_{,k} - 2r_{,i} r_{,n} r_{,k} - n_k\left(\mathbf{x}\right) r_{,i}}{2\pi r} d\Gamma = -\phi_{,k}\left(\mathbf{z}\right) \int_{\Gamma_\varepsilon} \frac{2n_i n_k}{2\pi r} d\Gamma =$$

$$-\phi_{,k}\left(\mathbf{z}\right) \lim_{\varepsilon \to 0} \left(\int_\theta \frac{2n_i n_k}{2\pi \varepsilon} \varepsilon d\theta\right) = -\phi_{,k}\left(\mathbf{z}\right) \int_\theta \frac{n_i n_k}{\pi} d\theta = -\phi_{,k}\left(\mathbf{z}\right) \frac{\delta_{ik}}{2} = -\frac{\phi_{,i}\left(\mathbf{z}\right)}{2}$$

$$\text{(A.56)}$$

In the case of the integrals over the general boundary it is also necessary to analyse the behaviour close to the singularity. In this case (assuming smooth boundary) $d\Gamma \sim dr$.

Term $\boxed{T_4}$

$$\int_{\Gamma - \Gamma_\varepsilon} \frac{n_i\left(\mathbf{x}\right)}{2\pi r^2} \phi\left(\mathbf{x}\right) d\Gamma = \overbrace{\int_{\Gamma - \Gamma_\varepsilon} \frac{n_i\left(\mathbf{x}\right)}{2\pi r^2} \left[\phi\left(\mathbf{x}\right) - \phi\left(\mathbf{z}\right) - \phi_{,k}\left(\mathbf{z}\right) r_k\right] d\Gamma}^{A_4}$$

$$\text{(A.57)}$$

$$+ \overbrace{\phi\left(\mathbf{z}\right) \int_{\Gamma - \Gamma_\varepsilon} \frac{n_i\left(\mathbf{x}\right)}{2\pi r^2} d\Gamma}^{A_5} + \overbrace{\phi_{,k}\left(\mathbf{z}\right) \int_{\Gamma - \Gamma_\varepsilon} \frac{n_i\left(\mathbf{x}\right) r_k}{2\pi r^2} d\Gamma}^{A_6}$$

Remembering that

$$\boldsymbol{\nabla}_s \phi = \boldsymbol{\nabla}\phi - \mathbf{n}\left(\mathbf{n}\boldsymbol{\nabla}\phi\right) \Longrightarrow \phi_{,k} = \phi^S_{,k} + n_k q \qquad \text{(A.58)}$$

$$\lim_{\varepsilon \to 0}\left(\int_{\Gamma - \Gamma_\varepsilon} \mathrm{I}\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma\right) = \fint_\Gamma \mathrm{I}\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma \qquad \text{(A.59)}$$

and taking into account that if $\mathrm{I}\left(\mathbf{x};\mathbf{z}\right)$ is weakly singular in Γ

$$\fint_\Gamma \mathrm{I}\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma = \int_\Gamma \mathrm{I}\left(\mathbf{x};\mathbf{z}\right)\mathrm{d}\Gamma \qquad \text{(A.60)}$$

Term $\boxed{\mathrm{A}_4}$

$$\int_{\Gamma - \Gamma_\varepsilon} \frac{n_i\left(\mathbf{x}\right)}{2\pi r^2}\left[\phi\left(\mathbf{x}\right) - \phi\left(\mathbf{z}\right) - \phi_{,k}\left(\mathbf{z}\right)r_k\right]\mathrm{d}\Gamma =$$

$$\overbrace{\int_\Gamma \frac{n_i\left(\mathbf{x}\right)}{2\pi r^2}\left[\phi\left(\mathbf{x}\right) - \phi\left(\mathbf{z}\right) - \phi^S_{,k}\left(\mathbf{z}\right)r_k\right]\mathrm{d}\Gamma}^{\text{weakly singular}} - n_k\left(\mathbf{z}\right)q\left(\mathbf{z}\right)\overbrace{\int_\Gamma \frac{n_i\left(\mathbf{x}\right)r_{,k}}{2\pi r}\mathrm{d}\Gamma}^{\text{weakly singular}} \qquad \text{(A.61)}$$

where it has been considered that $n_k r_{,k} \to 0$ for $\mathbf{x} \to \mathbf{z}$.

At the same time

Term $\boxed{\text{-}\mathrm{T}_5}$

$$-\int_{\Gamma - \Gamma_\varepsilon} \frac{2r_{,i}r_{,n}}{2\pi r^2}\phi\left(\mathbf{x}\right)\mathrm{d}\Gamma = -\overbrace{\int_\Gamma \frac{2r_{,i}r_{,n}}{2\pi r^2}\left[\phi\left(\mathbf{x}\right) - \phi\left(\mathbf{z}\right)\right]\mathrm{d}\Gamma}^{\text{weakly singular}} \qquad \text{(A.62)}$$

$$-\phi\left(\mathbf{z}\right)\overbrace{\int_{\Gamma - \Gamma_\varepsilon} \frac{2r_{,i}r_{,n}}{2\pi r^2}\mathrm{d}\Gamma}^{\mathrm{A}_7}$$

The next step is to combine A_5 and A_7 to get

Term $\boxed{A_5 - A_7}$

$$\phi\left(\mathbf{z}\right)\int_{\Gamma-\Gamma_\varepsilon}\frac{n_i\left(\mathbf{x}\right)-2r_ir_{,n}}{2\pi r^2}\mathrm{d}\Gamma=\phi\left(\mathbf{z}\right)\int_\Gamma\overbrace{\frac{n_i\left(\mathbf{x}\right)-n_i\left(\mathbf{z}\right)\left|\frac{\mathrm{d}r}{\mathrm{d}s}\right|-2r_{,i}r_{,n}}{2\pi r^2}}^{\text{weakly singular}}\mathrm{d}\Gamma+$$

$$\overbrace{\phi\left(\mathbf{z}\right)n_i\left(\mathbf{z}\right)\int_{\Gamma-\Gamma_\varepsilon}\frac{1}{2\pi r^2}\left|\frac{\mathrm{d}r}{\mathrm{d}s}\right|\mathrm{d}\Gamma}^{A_8}$$

$$\text{(A.63)}$$

where it has been considered that

$$n_i\left(\mathbf{x}\right)\approx n_i\left(\mathbf{z}\right)-\kappa\left(\mathbf{z}\right)r_i+O\left(r\right)^2$$

$$r_{,n}\left(\mathbf{x}\right)\approx-\frac{\kappa\left(\mathbf{z}\right)}{2}r+O\left(r\right)^2 \qquad\text{(A.64)}$$

$$\frac{\mathrm{d}r}{\mathrm{d}s}\approx\pm1+O\left(r\right)^2$$

Term $\boxed{A_8}$

When solving A_8 some assumptions are introduced in order to simplify the resultant equation. Formally A_8 can be divided in two intervals of integration. The first one close to the singularity Γ_c and the second one including the remainder boundary Γ_r as it is shown in figure (A.3).
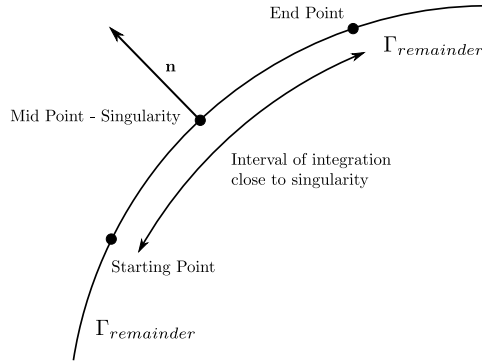
Figure A.3: Decomposition of the boundary

So

$$
\phi\left(\mathbf{z}\right) n_i\left(\mathbf{z}\right) \int_{\Gamma-\Gamma_\varepsilon} \frac{1}{2\pi r^2} \left|\frac{\mathrm{d}r}{\mathrm{d}s}\right| \mathrm{d}\Gamma = \phi\left(\mathbf{z}\right) n_i\left(\mathbf{z}\right) \int_{\Gamma_c-\Gamma_\varepsilon} \frac{1}{2\pi r^2} \left|\frac{\mathrm{d}r}{\mathrm{d}s}\right| \mathrm{d}\Gamma +
$$

$$
\phi\left(\mathbf{z}\right) n_i\left(\mathbf{z}\right) \int_{\Gamma_r} \frac{1}{2\pi r^2} \left|\frac{\mathrm{d}r}{\mathrm{d}s}\right| \mathrm{d}\Gamma
$$

$\boxed{\text{A.65}}$

In practice, the previous equation are not used in the general case, but close to the singularity. In order to simplify it, the assumption $\Gamma_c = \Gamma$ is used, which is the actual implementation of the equation in a numerical method.

Then, the integral has been divided into two parts[2] due to the absolute value and can be solved analytically obtaining and unbounded term I_1 which can be cancelled with the corresponding term from equation (A.54). So

---

[2] The letters s,m and e refers, respectively to the starting, mid and end point of the curve.

$$\phi(\mathbf{z})\,n_i(\mathbf{z})\int_{\Gamma-\Gamma_\varepsilon}\frac{1}{2\pi r^2}\left|\frac{\mathrm{d}r}{\mathrm{d}s}\right|\mathrm{d}\Gamma = \phi(\mathbf{z})\,n_i(\mathbf{z})\overbrace{\int_m^s\frac{1}{2\pi r^2}\mathrm{d}r}^{\left|\frac{\mathrm{d}r}{\mathrm{d}s}\right|<0}+$$

$$\text{(A.66)}$$

$$\phi(\mathbf{z})\,n_i(\mathbf{z})\overbrace{\int_m^e\frac{1}{2\pi r^2}\mathrm{d}r}^{\left|\frac{\mathrm{d}r}{\mathrm{d}s}\right|>0} = -\frac{\phi(\mathbf{z})\,n_i(\mathbf{z})}{2\pi}\left[\frac{1}{r_s}+\frac{1}{r_e}\right]+\overbrace{\frac{\phi(\mathbf{z})\,n_i(\mathbf{z})}{\pi\varepsilon}}^{I_1}$$

If we analyse the last term

Term $\boxed{-T_6}$

$$-\int_{\Gamma-\Gamma_\varepsilon}\frac{r_{,i}}{2\pi r}q(\mathbf{x})\,\mathrm{d}\Gamma = -\overbrace{\int_\Gamma\frac{r_{,i}}{2\pi r}\left[q(\mathbf{x})-\phi_{,k}(\mathbf{z})\,n_k(\mathbf{x})\right]\mathrm{d}\Gamma}^{A_9}$$

$$\text{(A.67)}$$

$$-\phi_{,k}(\mathbf{z})\overbrace{\int_{\Gamma-\Gamma_\varepsilon}\frac{r_{,i}n_k(\mathbf{x})}{2\pi r}\mathrm{d}\Gamma}^{A_{10}}$$

Term $\boxed{-A_9}$

$$-\int_\Gamma\frac{r_{,i}}{2\pi r}\left[q(\mathbf{x})-\phi_{,k}(\mathbf{z})\,n_k(\mathbf{x})\right]\mathrm{d}\Gamma = -\overbrace{\int_\Gamma\frac{r_{,i}}{2\pi r}\left[q(\mathbf{x})-q(\mathbf{z})\right]\mathrm{d}\Gamma}^{\text{weakly singular}}$$

$$\text{(A.68)}$$

$$-\phi_{,k}(\mathbf{z})\overbrace{\int_\Gamma\frac{r_{,i}}{2\pi r}\left[n_k(\mathbf{z})-n_k(\mathbf{x})\right]\mathrm{d}\Gamma}^{\text{weakly singular}}$$

and if we combine A_8 and A_{10}

APPENDIX A. KERNELS AND LIMITS

Term $\boxed{A_6 - A_{10}}$

$$\phi_{,k}\left(\mathbf{z}\right) \int_{\Gamma-\Gamma_\varepsilon} \frac{r_{,k} n_i\left(\mathbf{x}\right) - r_{,i} n_k\left(\mathbf{x}\right)}{2\pi r} \mathrm{d}\Gamma = \epsilon_{ik}\phi_{,k}\left(\mathbf{z}\right) \int_{\Gamma-\Gamma_\varepsilon} \frac{r_{,2} n_1 - r_{,1} n_2\left(\mathbf{x}\right)}{2\pi r} \mathrm{d}\Gamma =$$

$$-\epsilon_{ik}\phi_{,k}\left(\mathbf{z}\right) \int_{\Gamma-\Gamma_\varepsilon} \frac{r_{,l} t_l\left(\mathbf{x}\right)}{2\pi r} \mathrm{d}\Gamma = -\epsilon_{ik}\phi_{,k}\left(\mathbf{z}\right) \int_{\Gamma-\Gamma_\varepsilon} \frac{\boldsymbol{\nabla} r}{2\pi} \mathrm{d}\mathbf{l} =$$

$$-\epsilon_{ik}\phi_{,k}\left(\mathbf{z}\right) \int_s^m \frac{\boldsymbol{\nabla} r}{2\pi} \mathrm{d}\mathbf{l} - \epsilon_{ik}\phi_{,k}\left(\mathbf{z}\right) \int_m^e \frac{\boldsymbol{\nabla} r}{2\pi} \mathrm{d}\mathbf{l} =$$

$$-\frac{\epsilon_{ik}\phi_{,k}\left(\mathbf{z}\right)}{2\pi} \left[\log\left(r_e\right) - \log\left(r_s\right)\right]$$

$$\boxed{\text{A.69}}$$

where the Levi-civita symbol ϵ_{ij} has been introduced to compact the equation and the procedure used with the A_8 term has been followed again.

So reordering the terms we gete

$$\frac{1}{2}\phi_{,i}\left(\mathbf{z}\right) - \phi_{,k}\left(\mathbf{z}\right) \int_\Gamma \frac{r_{,i}}{2\pi r} \left[n_k\left(\mathbf{z}\right) - n_k\left(\mathbf{x}\right)\right] \mathrm{d}\Gamma - \frac{\epsilon_{ik}\phi_{,k}\left(\mathbf{z}\right)}{2\pi} \left[\log\left(r_e\right) - \log\left(r_s\right)\right] +$$

$$\int_\Gamma \frac{n_i\left(\mathbf{x}\right)}{2\pi r^2} \left[\phi\left(\mathbf{x}\right) - \phi\left(\mathbf{z}\right) - \phi_{,k}^s\left(\mathbf{z}\right) r_k\right] \mathrm{d}\Gamma - \int_\Gamma \frac{2 r_{,i} r_{,n}}{2\pi r^2} \left[\phi\left(\mathbf{x}\right) - \phi\left(\mathbf{z}\right)\right] \mathrm{d}\Gamma +$$

$$\phi\left(\mathbf{z}\right) \int_\Gamma \frac{n_i\left(\mathbf{x}\right) - n_i\left(\mathbf{z}\right) \left|\frac{\mathrm{d}r}{\mathrm{d}s}\right| - 2 r_{,i} r_{,n}}{2\pi r^2} \mathrm{d}\Gamma - \frac{\phi\left(\mathbf{z}\right) n_i\left(\mathbf{z}\right)}{2\pi} \left[\frac{1}{r_s} + \frac{1}{r_e}\right] =$$

$$\int_\Gamma \frac{r_{,i}}{2\pi r} \left[q\left(\mathbf{x}\right) - q\left(\mathbf{z}\right)\right] \mathrm{d}\Gamma + n_k\left(\mathbf{z}\right) q\left(\mathbf{z}\right) \int_\Gamma \frac{n_i\left(\mathbf{x}\right) r_{,k}}{2\pi r} \mathrm{d}\Gamma$$

$$\boxed{\text{A.70}}$$

Bibliography

[1] A.M. Afsar and J. Go. Finite element analysis of thermoelastic field in a rotating FGM circular disk. *Applied Mathematical Modelling*, 34(11):3309–3320, 2010.

[2] E. Akin. *Object-Oriented Programming Via Fortran 90/95*. Cambridge University Press, 2003.

[3] A. Aleynikov. *Spatial Contact Problems in Geotechnics: Boundary-Element Method*. Springer, 2010.

[4] A. Ali and C. Rajakumar. *The boundary element method: applications in sound and vibration*. Taylor Francis Routledge, 2004.

[5] H. Aliabadi. *The Boundary Element Method Volume 2: Applications in Solids and Structures*. Wiley, 2002.

[6] W.-T. Ang, D.L. Clements, and N. Vahdati. A dual-reciprocity boundary element method for a class of elliptic boundary value problems for non-homogeneous anisotropic media. *Engineering Analysis with Boundary Elements*, 27(1):49–55, 2003.

[7] W.T. Ang, J. Kusuma, and D.L. Clements. A boundary element method for a second order elliptic partial differential equation with variable coefficients. *Engineering Analysis with Boundary Elements*, 18(4):311–316, 1996.

[8] W. Annicchiarico, G. Martinez, and M. Cerrolaza. Boundary elements and B-spline surface modeling for medical applications. *Applied Mathematical Modelling*, 31(2):194–208, 2007.

[9] D.N. Arnold and Wendland W.L. Collocation versus Galerkin procedures for boundary integral methods. In *Boundary Element Methods in Engineering*, 1982.

[10] S. N. Atluri and T. Zhu. A new Meshless Local Petrov-Galerkin (MLPG) approach in computational mechanics. *Computational Mechanics*, 22(2):117–127, 1998.

[11] P.K. Banerjee and R. Butterfield. *Boundary element methods in engineering science.* McGraw-Hill Book Co. (UK), 1981.

[12] M. Bebendorf. *Hierarchical Matrices: A Means to Efficiently Solve Elliptic Boundary Value Problems.* Springer, 2008.

[13] M. Bebendorf and R. Grzhibovskis. Accelerating Galerkin BEM for linear elasticity using adaptive cross approximation. *Mathematical Methods in the Applied Sciences*, 29(14):1721–1747, 2006.

[14] A.A. Becker. *The boundary element method in engineering: a complete course.* McGraw-Hill, 1992.

[15] J.R. Berger, P.A. Martin, V. Mantič, and L.J. Gray. Fundamental solutions for steady-state heat transfer in an exponentially graded anisotropic material. *Zeitschrift fur Angewandte Mathematik und Physik*, 56(2):293–303, 2005.

[16] J. Besson and R. Foerch. Large scale object-oriented finite element code design. *Computer Methods in Applied Mechanics and Engineering*, 142(1-2):165–187, 1997.

[17] V. Birman and L.W. Byrd. Modeling and analysis of functionally graded materials and structures. *Applied Mechanics Reviews*, 60(1-6):195–216, 2007.

[18] C.A. Brebbia. *The Boundary Element Method For Engineers.* Pentech Press, 1984.

[19] C.A. Brebbia and J. Dominguez. *Boundary Elements: An introductory course (second edition).* Computational Mechanics Publications, 1992.

[20] M.D. Buhmann. *Radial Basis Functions: Theory and Implementations.* Cambridge Univeristy Press, 2003.

[21] J.R. Bunch and J.E. Hopcroft. Triangular Factorization and Inversion by Fast Matrix Multiplication. *Mathematics of Computation*, 28(125):231–236, 1974.

[22] Y.-S. Chan, L.J. Gray, T. Kaplan, and G.H. Paulino. Green's function for a two-dimensional exponentially graded elastic medium. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 460(2046):1689–1706, 2004.

[23] C.S. Chen, C.A. Brebbia, and H. Power. Dual reciprocity method using compactly supported radial basis functions. *Communications in Numerical Methods in Engineering*, 15(2):137–150, 1999.

[24] T. Chen, B. Wang, Z. Cen, and Z. Wu. A symmetric Galerkin multizone boundary element method for cohesive crack growth. *Engineering Fracture Mechanics*, 63(5):591–609, 1999.

[25] W. Chen and M. Tanaka. A meshless, integration-free, and boundary-only RBF technique. *Computers and Mathematics with Applications*, 43(3-5):379–391, 2002.

[26] A.H.-D. Cheng and D.T. Cheng. Heritage and early history of the boundary element method. *Engineering Analysis with Boundary Elements*, 29(3):268–302, 2005.

[27] Alexander H.D. Cheng. Darcy's Flow With Variable Permeability: A Boundary Integral Solution. *Water Resources Research*, 20(7):980–984, 1984.

[28] C.L. Clements. A fundamental solution for linear second-order elliptic systems with variable coefficients. *Journal of Engineering Mathematics*, 49(4):209–216, 2004.

[29] D.L. Clements. Fundamental solutions for second order linear elliptic partial differential equations. *Computational Mechanics*, 22(1):26–31, 1998.

[30] R. Criado, L. J. Gray, V. Mantič, and F. París. Green's function evaluation for three-dimensional exponentially graded elasticity. *International Journal for Numerical Methods in Engineering*, 74(10):1560–1591, 2008.

[31] J. M. Crotty. A block equation solver for large unsymmetric matrices arising in the boundary integral equation method. *International Journal for Numerical Methods in Engineering*, 18(7):997–1017, 1982.

[32] M. Dehghan and A. Ghesmati. Application of the dual reciprocity boundary integral equation technique to solve the nonlinear Klein-Gordon equation. *Computer Physics Communications*, 181(8):1410–1418, 2010.

[33] M. Dehghan and A. Ghesmati. Application of the dual reciprocity boundary integral equation technique to solve the nonlinear Klein-Gordon equation. *Computer Physics Communications*, 181(8):1410–1418, 2010.

[34] J. Dominguez. *Boundary Elements in Dynamics*. Computational Mechanics Publications, 1993.

[35] J. Dominguez, M.P. Ariza, and R. Gallego. Flux and traction boundary elements without hypersingular or strongly singular integrals. *International Journal for Numerical Methods in Engineering*, 48(1):111–135, 2000.

[36] M.A. Fahmy. A time-stepping DRBEM for magneto-thermo-viscoelastic interactions in a rotating nonhomogeneous anisotropic solid. *International Journal of Applied Mechanics*, 3(4):711–734, 2011.

[37] G. Fairweather and A. Karageorghis. The method of fundamental solutions for elliptic boundary value problems. *Advances in Computational Mathematics*, 9(1-2):69–95, 1998.

[38] C. Franke and R. Schaback. Solving partial differential equations by collocation using radial basis functions. *Applied Mathematics and Computation*, 93(1):73–82, 1998.

[39] R. Gallego and Dominguez J. A Two Dimensional Boundary Element Code for Time-Domain Formulations Using Quadratic Elements: I Potential Problems. In *Boundary element communications*, volume 5(4), pages 117–125, 1998.

[40] X. Gao and K. Yang. Thermal stress analysis of functionally graded material structures using boundary element method. *Lixue Xuebao/Chinese Journal of Theoretical and Applied Mechanics*, 43(1):136–143, 2011.

[41] X.-W. Gao. The radial integration method for evaluation of domain integrals with boundary-only discretization. *Engineering Analysis with Boundary Elements*, 26(10):905–916, 2002.

[42] X.-W. Gao, L. Guo, and Ch. Zhang. Three-step multi-domain BEM solver for nonhomogeneous material problems. *Engineering Analysis with Boundary Elements*, 31(12):965–973, 2007.

[43] X.W. Gao, Ch. Zhang, J. Sladek, and V. Sladek. Fracture analysis of functionally graded materials by a BEM. *Composites Science and Technology*, 68(5):1209–1215, 2008.

[44] N.G. Gençer and I.O. Tanzer. Forward problem solution of electromagnetic source imaging using a new BEM formulation with high-order elements. *Physics in Medicine and Biology*, 44(9):2275–2287, 1999.

[45] M.A. Golberg, C.S. Chen, and H. Bowman. Some recent results and proposals for the use of radial basis functions in the BEM. *Engineering Analysis with Boundary Elements*, 23(4):285–296, 1999.

[46] A. M. Gorelik. Object-Oriented Programming in Modern Fortran. *Programming and Computer Software*, 30(3):173–179, 2004.

[47] E. Graciani, V. Mantič, F. París, and J. Cañas. Critical study of hypersingular and strongly singular boundary integral representations of potential gradient. *Computational Mechanics*, 25(6):542–559, 2000.

[48] L.J. Gray, T. Kaplan, J.D. Richardson, and G.H. Paulino. Green's Functions and Boundary Integral Analysis for Exponentially Graded Materials: Heat Conduction. *Journal of Applied Mechanics, Transactions ASME*, 70(4):543–549, 2003.

[49] A. Gupta, S. Chempath, M.J. Sanborn, L.A. Clark, and R.Q. Snurr. Object-oriented programming paradigms for molecular modeling. *Molecular Simulation*, 29(1):29–46, 2003.

[50] W.S. Hall and G. Oliveto. *Boundary Element Methods for Soil-Structure Interaction*. Kluwer Academic Pub, 2003.

[51] ISO. *ISO/IEC 1539-1:2004 Information technology. Programming languages: Fortran Part 1: Base language*. pub-IEC, 2004.

[52] D.K. Jha, T. Kant, and R.K. Singh. A critical review of recent research on functionally graded plates. *Composite Structures*, 96:833–849, 2013.

[53] J.H. Kane, B.L. Kashava Kumar, and S. Saigal. An arbitrary condensing, noncondensing solution strategy for large scale, multi-zone boundary element analysis. *Computer Methods in Applied Mechanics and Engineering*, 79(2):219–244, 1990.

[54] E. J. Kansa. Multiquadrics-A scattered data approximation scheme with applications to computational fluid-dynamics-II solutions to parabolic, hyperbolic and elliptic partial differential equations. *Computers and Mathematics with Applications*, 19(8-9):147–161, 1990.

[55] S.R. Karur and P.A. Ramachandran. Radial basis function approximation in the dual reciprocity method. *Mathematical and Computer Modelling*, 20(7):59–70, 1994.

[56] Sriganesh R. Karur and P.A. Ramachandran. Augmented thin plate spline approximation in DRM. *Boundary elements communications*, 6(2):55–58, 1995. cited By (since 1996)26.

[57] J. T. Katsikadelis. The analog equation method - a powerful BEM-based solution technique for solving linear and nonlinear engineering

problems. *Computational Mechanics Publications - Boundary Element Method XVI*, 1(16):167–182, 1994.

[58] J.T. Katsikadelis. The analog equation method—a powerful BEM-based solution technique for solving linear and nonlinear engineering problems. In *Boundary element method XVI*, 1994.

[59] J.T. Katsikadelis. Dynamic analysis of nonlinear membranes by the analog equation method: A boundary-only solution. *Computational Mechanics*, 29(2):170–177, 2002.

[60] J.T. Katsikadelis. The BEM for nonhomogeneous bodies. *Archive of Applied Mechanics*, 74(11-12):780–789, 2005.

[61] J.T. Katsikadelis. The 2D elastostatic problem in inhomogeneous anisotropic bodies by the meshless analog equation method (MAEM). *Engineering Analysis with Boundary Elements*, 32(12):997–1005, 2008.

[62] J.T. Katsikadelis and C.B. Kandilas. Solving the plane elastostatic problem by the analog equation method. *Computers and Structures*, 64(1-4):305–312, 1997. cited By (since 1996)4.

[63] J.T. Katsikadelis and M.S. Nerantzaki. Boundary element method for nonlinear problems. *Engineering Analysis with Boundary Elements*, 23(5):365–373, 1999.

[64] J.-H. Kim and G.H. Paulino. Isoparametric graded finite elements for nonhomogeneous isotropic and orthotropic materials. *Journal of Applied Mechanics, Transactions ASME*, 69(4):502–514, 2002.

[65] H.-Y. Kuo and T. Chen. Steady and transient Green's functions for anisotropic conduction in an exponentially graded solid. *International Journal of Solids and Structures*, 42(3-4):1111–1128, 2005.

[66] C. Lage. The application of object-oriented methods to boundary elements. *Computer Methods in Applied Mechanics and Engineering*, 157(3-4):205–213, 1998.

[67] J.B. Layton, S. Ganguly, C. Balakrishna, and J.H. Kane. A symmetric galerkin multi-zone boundary element formulation. *International Journal for Numerical Methods in Engineering*, 40(16):2913–2931, 1997.

[68] S.-J. Liao. High-order BEM formulations for strongly non-linear problems governed by quite general non-linear differential operators. *International Journal for Numerical Methods in Fluids*, 23(8):739–751, 1996.

[69] S.-J. Liao. Boundary element method for general nonlinear differential operators. *Engineering Analysis with Boundary Elements*, 20(2):91–99, 1997.

[70] S.J. Liao. The quite general BEM for strongly non-linear problems. *Boundary Elements XVII*, pages 67–74, 1995. cited By (since 1996)11.

[71] Y.J. Liu and N. Nishimura. The fast multipole boundary element method for potential problems: A tutorial. *Engineering Analysis with Boundary Elements*, 30(5):371–381, 2006.

[72] J. Mackerle. Object-oriented programming in FEM and BEM: A bibliography (1990-2003). *Advances in Engineering Software*, 35(6):325–336, 2004.

[73] G.D. Manolis and R.P. Shaw. Green's function for the vector wave equation in a mildly heterogeneous continuum. *Wave Motion*, 24(1):59–83, 1996.

[74] R.J. Marczak. An object-oriented programming framework for boundary integral equation methods. *Computers and Structures*, 82(15–16):1237–1257, 2004.

[75] L. Marin, L. Elliott, P.J. Heggs, D.B. Ingham, D. Lesnic, and X. Wen. Dual reciprocity boundary element method solution of the Cauchy problem for Helmholtz-type equations with variable coefficients. *Journal of Sound and Vibration*, 297(1-2):89–105, 2006.

[76] L. Marin, L. Elliott, P.J. Heggs, D.B. Ingham, D. Lesnic, and X. Wen. Dual reciprocity boundary element method solution of the Cauchy problem for Helmholtz-type equations with variable coefficients. *Journal of Sound and Vibration*, 297(1-2):89–105, 2006.

[77] P.A. Martin, J.D. Richardson, L.J. Gray, and J.R. Berger. On Green's function for a three-dimensional exponentially graded elastic solid. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 458(2024):1931–1947, 2002.

[78] P.A. Martin and F.I. Rizzo. Hypersingular integrals: How smooth must the density be? *International Journal for Numerical Methods in Engineering*, 39(4):687–704, 1996.

[79] P.A. Martin, F.J. Rizzo, and T.A. Cruse. Smoothness-relaxation strategies for singular and hypersingular integral equations. *International Journal for Numerical Methods in Engineering*, 42(5):885–906, 1998.

[80] M. Metcalf, J.K. Reid, and M. Cohen. *Fortran 95/2003 Explained*. Oxford University Press, 2004.

[81] A.R. Mitchell. *The finite difference method in partial differential equations*. John Wiley and Sons, 1980.

[82] D. Nardini and C. A. Brebbia. A new approach to free vibration analysis using boundary elements. *Applied Mathematical Modelling*, 7(3):157–162, 1983.

[83] B. Natalini and V. Popov. Tests of radial basis functions in the 3D DRM-MD. *Communications in Numerical Methods in Engineering*, 22(1):13–22, 2006.

[84] B. Natalini and V. Popov. On the optimal implementation of the boundary element dual reciprocity method-multi-domain approach for 3D problems. *Engineering Analysis with Boundary Elements*, 31(3):275–287, 2007.

[85] M.S. Nerantzaki and N.G. Babouskos. Analysis of inhomogeneous anisotropic viscoelastic bodies described by multi-parameter fractional differential constitutive models. *Computers and Mathematics with Applications*, 62(3):945–960, 2011. cited By (since 1996)4.

[86] M.S. Nerantzaki and C.B. Kandilas. A boundary element method solution for anisotropic nonhomogeneous elasticity. *Acta Mechanica*, 200(3-4):199–211, 2008. cited By (since 1996)4.

[87] M.S. Nerantzaki and J.T. Katsikadelis. Buckling of plates with variable thickness - An analog equation solution. *Engineering Analysis with Boundary Elements*, 18(2):149–154, 1996.

[88] M.S. Nerantzaki and J.T. Katsikadelis. Nonlinear dynamic analysis of circular plates with varying thickness. *Archive of Applied Mechanics*, 77(6):381–391, 2007.

[89] N. Nishimura. Fast multipole accelerated boundary integral equation methods. *Applied Mechanics Reviews*, 55(4):299–324, 2002. cited By (since 1996)226.

[90] A.J. Nowak and A.C. Neves. *The Multiple reciprocity boundary element method*. Computational Mechanics Publications, 1994.

[91] P.W. Partridge. Approximation functions in the dual reciprocity method. *Boundary elements communications*, 8(1):1–4, 1997.

[92] P.W. Partridge. Towards criteria for selecting approximation functions in the Dual Reciprocity Method. *Engineering Analysis with Boundary Elements*, 24(7-8):519–529, 2000.

[93] P.W. Partridge, C.A. Brebbia, and L.C. Wrobel. *The Dual Reciprocity Boundary Element Method*. Computational Mechanics Publications, 1992.

[94] G.H. Paulino and A. Sutradhar. The simple boundary element method for multiple cracks in functionally graded media governed by potential

theory: A three-dimensional Galerkin approach. *International Journal for Numerical Methods in Engineering*, 65(12):2007–2034, 2006.

[95] H.-F. Peng, Y.-G. Bai, K. Yang, and X.-W. Gao. Three-step multi-domain BEM for solving transient multi-media heat conduction problems. *Engineering Analysis with Boundary Elements*, 37(11):1545–1555, 2013.

[96] H. Qiao. Object-oriented programming for the boundary element method in two-dimensional heat transfer analysis. *Advances in Engineering Software*, 37(4):248–259, 2006. cited By (since 1996)3.

[97] R.H. Rigby and M.H. Aliabadi. Out-of-core solver for large, multi-zone boundary element matrices. *International Journal for Numerical Methods in Engineering*, 38(9):1507–1533, 1995.

[98] M.A. Riveiro and R. Gallego. Boundary elements and the analog equation method for the solution of elastic problems in 3-D non-homogeneous bodies. *Computer Methods in Applied Mechanics and Engineering*, 263:12–19, 2013.

[99] S. Rjasanow and O. Steinbach. *The Fast Solution of Boundary Integral Equations*. Springer, 2007.

[100] E. Ruocco and V. Minutolo. Two-dimensional stress analysis of multi-region functionally graded materials using a field boundary element model. *Composites Part B: Engineering*, 43(2):663–672, 2012.

[101] Y Saad and M.H. Schultz. GMRES - A generalized minimal residual algorithm for solving nonsymmetric linear-systems. *Siam Journal on scientific and statistical computing*, 7(3):856–869, 1986.

[102] B.V. Sankar. An elasticity solution for functionally graded beams. *Composites Science and Technology*, 61(5):689–696, 2001.

[103] E.J. Sapountzakis. Dynamic analysis of composite steel-concrete structures with deformable connection. *Computers and Structures*, 82(9-10):717–729, 2004.

[104] R.P. Shaw. Green's functions for heterogeneous media potential problems. *Engineering Analysis with Boundary Elements*, 13(3):219–221, 1994.

[105] R.P. Shaw and N. Makris. Green's functions for Helmholtz and Laplace equations in heterogeneous media. *Engineering Analysis with Boundary Elements*, 10(2):179–183, 1992.

[106] N.D. Stringfellow, R.N.L. Smith, and V.V.S.S Sastry. A generic structure for scalar and vector boundary element codes using Fortran 90. *Advances in Engineering Software*, 30(5):313–325, 1999.

[107] A. Sutradhar and G.H. Paulino. A simple boundary element method for problems of potential in non-homogeneous media. *International Journal for Numerical Methods in Engineering*, 60(13):2203–2230, 2004.

[108] A. Sutradhar, G.H. Paulino, and L.J. Gray. Transient heat conduction in homogeneous and non-homogeneous materials by the Laplace transform Galerkin boundary element method. *Engineering Analysis with Boundary Elements*, 26(2):119–132, 2002.

[109] A. Sutradhar, G.H. Paulino, and L.J. Gray. *Symmetric Galerkin Boundary Element Method.* Springer, 2008.

[110] Alok Sutradhar and Glaucio H. Paulino. The simple boundary element method for transient heat conduction in functionally graded materials. *Computer Methods in Applied Mechanics and Engineering*, 193(42–44):4511 – 4539, 2004.

[111] M. Tanaka, T. Matsumoto, and Y. Suda. A dual reciprocity boundary element method applied to the steady-state heat conduction problem of functionally gradient materials (study on two-dimensional problems). *Nihon Kikai Gakkai Ronbunshu, A Hen/Transactions of the Japan Society of Mechanical Engineers, Part A*, 67(662):1589–1594, 2001.

[112] M. Tanaka, T. Matsumoto, and S. Takakuwa. Dual reciprocity BEM for time-stepping approach to the transient heat conduction problem

in nonlinear materials. *Computer Methods in Applied Mechanics and Engineering*, 195(37-40):4953–4961, 2006.

[113] M. Tanaka, T. Matsumoto, and S. Takakuwa. Dual reciprocity BEM for time-stepping approach to the transient heat conduction problem in nonlinear materials. *Computer Methods in Applied Mechanics and Engineering*, 195(37-40):4953–4961, 2006.

[114] R. Vazquez, A. Buffa, and L. Di Rienzo. NURBS-Based BEM implementation of high-order surface impedance boundary conditions. *IEEE Transactions on Magnetics*, 48(12):4757–4766, 2012.

[115] H. Versteeg and W. Malalasekera. *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*. Research Studies Press ltd, 1995.

[116] H. Wang and Q.-H. Qin. Meshless approach for thermo-mechanical analysis of functionally graded materials. *Engineering Analysis with Boundary Elements*, 32(9):704–712, 2008.

[117] W. Wang, X. Ji, and Y. Wang. Object-oriented programming in boundary element methods using C++. *Advances in Engineering Software*, 30(2):127–132, 1999.

[118] L.C. Wrobel. *The boundary element method Volume 1: Applications in thermo-fluids and acoustics*. John Wiley and Sons, 2002.

[119] Y.Y. Wu, S.J. Liao, and X.Y. Zhao. Some notes on the general boundary element method for highly nonlinear problems. *Communications in Nonlinear Science and Numerical Simulation*, 10(7):725–735, 2005.

[120] K.H. Yu, A.H. Kadarman, and H. Djojodihardjo. Development and implementation of some BEM variants-A critical review. *Engineering Analysis with Boundary Elements*, 34(10):884–899, 2010.

[121] G. Yum, W.J. Yum, J.A.M. Carrer, and L. Gong. Stability of Galerkin and collocation time domain boundary element methods as applied to the scalar wave equation. *Computational Struct*, 74(4):495–506, 2000.

BIBLIOGRAPHY

[122] A.M. Zenkour. Generalized shear deformation theory for bending analysis of functionally graded plates. *Applied Mathematical Modelling*, 30(1):67–84, 2006.

[123] C. Zhang, M. Cui, J. Wang, X.W. Gao, J. Sladek, and V. Sladek. 3D crack analysis in functionally graded materials. *Engineering Fracture Mechanics*, 78(3):585–604, 2011.

[124] T. Zhu, J.-D. Zhang, and S.N. Atluri. A local boundary integral equation (LBIE) method in computational mechanics, and ameshless discretization approach. *Computational Mechanics*, 21(3):223–235, 1998.

[125] C. Zienkiewicz, R.L. Taylor, and J.Z. Zhu. *The Finite Element Method: Its Basis And Fundamentals (Sixth Edition)*. Butterworth-Heinemann, 2005.